ISBN: 978-3-95450-232-5 ISSN: 2673-7000 doi:10.18429/JACoW-NAPAC2022-WEPA70

TENSOR DECOMPOSITION FOR THE COMPRESSION AND ANALYSIS OF 10 kHz BPM DATA*

J. Choi†, Y. Hidaka, Y. Hu, G. Wang, BNL, Upton, NY 11973, USA

Abstract

In NSLS-II storage ring, during the user operation, fast-acquisition (FA) 10 kHz Beam Position Monitor (BPM) data are collected and their spectral properties are analyzed. Various periodograms and spectral peaks are being provided every minute, and they are very useful in identifying any changes in the orbit. Unfortunately, because of the big size of the data, only several numbers are being continually archived for the later study and the full raw data are saved only by hand when needed. We are developing methods utilizing tensor decomposition techniques to save and analyze the FA data and the paper is reporting the current status.

INTRODUCTION

As the user facility providing high-performance synchrotron light to the user, NSLS-II invests efforts constantly to maintain and improve the electron beam stability. As one of the efforts, we keep monitoring the 10 kHz BPM FA data properties. That is, every minute, we measure FA data for all BPMs for 10 seconds and their spectrum properties are displayed in the control system studio (CSS) pages. Because of the number of BPMs are more than 400 when we take account of the planes for each BPM, on top of the individual properties, representative spectrum properties are also displayed. The representative plots include averaged spectrum properties for dispersive region, non-dispersive region and ID BPMs.

Even though the plots are quite helpful and convenient tools to monitor the beam stability, all the FA data cannot be kept for the later use because of the big file size, and only some typical numbers are archived. Even though the archived numbers such as peak frequencies, power spectra are very useful, the contained information cannot be compared to the full FA data. Furthermore, the plots and the representative numbers are just being refreshed every minute, and picking up the moment when some variation is involved in beam stability is not easy.

In short, we searched the solution which compress the FA data to the reasonable size to keep all the files generated every minute and, at the same time, can provide several representative numbers showing the orbit status in real time. As a strong candidate, we tried the tensor decomposition analysis (TCA) which is well established in biology and medical society.

In the following sections, we introduce the TCA and shows how it is applied to NSLS-II storage ring FA data, together with the current status.

* Work supported by DOE under contract No. DE-SC0012704.
† jchoi@bnl.gov

TENSOR DECOMPOSITION

For the analysis of the matrices, principal component analysis (PCA) methods like singular value decomposition (SVD) or independent analysis (ICA) are well established and are playing critical roles in various fields of expertise. Similar techniques for high-dimensional array (tensor) are also well developed and being actively used in biology and medical society [1].

The main purpose of such analysis is identifying the limited number of factors which can explain the major part of the data. That means we can find low-rank representation of the tensor which can be used in compressing the data. The low-rank representation can reveal the factors behind the data. Also, we can have physical meanings for the identified factors.

There are several popular decomposition techniques [1], but here we choose CANDECOMP/PARAFAC (CP) decomposition. The CP decomposition is simple and small number of parameters can represent the tensor. Therefore, the method can compress the data efficiently when the system has clear low-rank behaviors.

Different from the biological or medical system, the physical system such as storage ring has a few clear physical parameters which dominate the beam behaviors. If we assume that the three-dimensional tensor $\mathcal{X}$ has R ranks, with the CP decomposition, it can be expressed as

$$\mathcal{X} \approx \sum_{r=1}^{R} u_r^{(1)} \circ u_r^{(2)} \circ u_r^{(3)} \qquad (1)$$

where u_r's are vectors for each dimension and $\circ$ is outer product. The schematic diagram of CP decomposition is shown in Fig. 1.

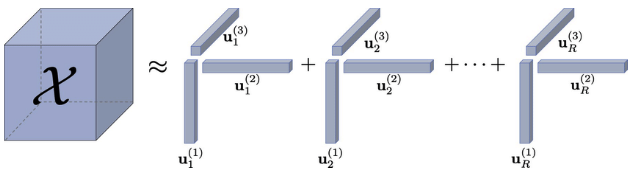

Figure 1: Diagram of CP decomposition (courtesy of Jean Kossaifi).

As in the case of matrix SVD, the magnitudes of u_r vectors are arbitrary as far as the products gives the correct tensor. Therefore, usually they are normalized and additional scale factors are introduced. Then, Eq. (1) can be represented as

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \times u_r^{(1)} \circ u_r^{(2)} \circ u_r^{(3)}. \qquad (2)$$

APPLICATION

As mentioned, NSLS-II generates FA data every minute for the analysis. These files cannot be archived but they are temporarily saved for about one hour. While they are available in the file system, we stack the data and make a 3-dimensional tensor. In fact, the FA data file is not generated every minute and the number of collected data sets in one hour is about 45. And if we collect for four hours, the number is around 180.

The FA data is 10 kHz and measured for 212 BPMs, considering that BPM measures the beam position in horizontal and vertical plane, the matrix size for a measurement is $100,000 \times 424$. Even though the data is measure for 10 seconds for the high resolution in the analysis, it is practically not possible to save and apply the TCA to the collection of the full data. For TCA, we collects the data corresponding 1 second, $10,000 \times 424$ matrices, periodically and the final tensor size becomes $\sim 10,000 \times 424 \times 140$ for 1-hr, and $\sim 10,000 \times 424 \times 180$ for 4-hr data collections.

If we decompose the tensor constructed by stacking the FA data, the u_r vectors correspoinding rank r in Eq. (1) or (2) can be interpreted as

- $u_r^{(1)}$: 10 kHz temporal vector (10,000 elements).
- $u_r^{(2)}$: Spatial vector around the ring (424 elements).
- $u_r^{(2)}$: Trend vector throughout the data acquisition (number of data sets).

We installed the script for the data collection and decomposition, and it is running periodically. The decomposition is performed only the desired number of data sets are collected. Once the desired number is collected, the tensor is decomposed by *rTensor* library [2] in *R* programming language. Here, we would like to note that, in rTensor, the u vectors in Eq. (2) are normalized with L1-norm. The CP decomposition algorithm requires the number of ranks in advance and we decompose the tensor with 1, 2, 3, and 10 as the number of ranks, which will be called as 1-rank, 2-ranks, 3-ranks, and 10-ranks decomposition in this paper.

The 1-rank, 2-ranks, and 3-ranks decompositions give information about how many ranks are involved during the given time span. As the measure of the number of ranks of the tensor, the closeness of the reconstructed tensor to the original tensor can be used. 10-ranks decomposition is included for the case full data is needed because when 10 is assigned as the number of ranks, it is almost always sure that the orginal tensor is reconstructed with more than 99% in whatever situation. In *rTensor*, the property is called *norm_percent* and expressed in Frobenius norm of tensor. That is *norm_percent* can be expressed as

$$norm\_percent = \left(1 - \frac{||\mathcal{X}_{reconstruct} - \mathcal{X}||}{||\mathcal{X}||}\right) \times 100. \quad (3)$$

In other words, *norm_percent* 100% means the compressed vectors can restore the original tensor perfectly.

At the first stage of the FA monitoring using the TCA, we used 4-hr data for the analysis. However, the computing time for the decomposition is too long and, more importantly, reviewing the stability every four hours is too late to respond to any variation. We reduce the time span to one hour and it turned out the time span is optimal in both aspects, the computing time and the reviewing period.

ANALYSIS

Since the TCA is running from June 2022, the typical *norm_percent* value for 1, 2, 3, and 10-ranks decompositions are shown in Table 1.

Table 1: Typical *Norm_Percent* Values When the Number of Ranks are Given as 1, 2, 3, and 10

ranks	**1**	**2**	**3**	**10**
norm_percent	99.35%	99.37%	99.37%	99.40%

Data Compression

As can be seen in Table 1, in most cases the tensor can be considered as rank-1 with noise. In this case we can compress the tensor of elements $10,000 \times 424 \times N$ to $10,000 + 424 + N$, with $N \sim 45$ or 180 depending on the data collection time span, one hour or four hours. In either case, the data size is compressed to less than 0.01% while the reconstructed tensor is more than 99% of the original tensor. Even when we save all the decompositions, ranks with 1, 2, 3, 10, the compressed size is less than 1% of the original tensor.

When the data is decomposed, all the *norm_percent*s are logged and we can detect the variations in the beam stability by these numbers because the difference in *norm_percent*s with different ranks provide the measure of the disturbance of the beam stability.

Since the TCA system is running in the NSLS-II storage ring, there were only three times when the *norm_percent* of 1-rank decomposition drops below 99%. Two of them are identified as BPM glitches and, in the other case, a very small variation in orbit spectrum could be identified.

These cases are described in the follwings.

Case 1

From the tensor data, collected during 2022-7-13 04:24 ~ 08:23, the resulting *norm_percent*s are shown as Table 2.

Table 2: *Norm_Percent* Values Logged in Case 1

ranks	**1**	**2**	**3**	**10**
norm_percent	68.26%	99.04%	99.04%	99.39%

From Table 3, we can see that the 1-rank decomposition is not enough to explain the orbit behavior, but it could be understood with 2-ranks decomposition. Because, different from matrix SVD, the λ values in Eq. (2) are neither guaranteed to be positive nor the vectors are orthogonal, some rearrangements are needed to identify which mode in 2-ranks corresponds to 1-rank decomposition. Figs. 2 and 3 shows the spatial and trend vectors of the 2-ranks decomposition vectors while the 1^{st} rank of 2-ranks vectors are almost equivalent to 1-rank decomposition vectors. It is not shown here, but the 2^{nd} rank temporal vector has non-physical spectrum.

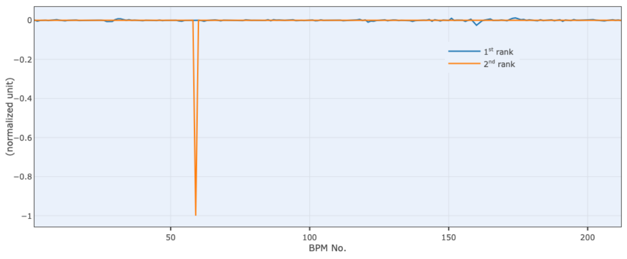

Figure 2: 2-ranks decomposition spatial vector for case 1.

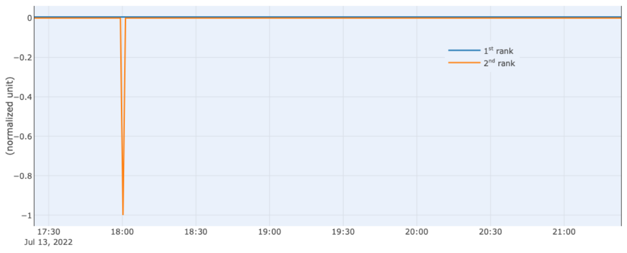

Figure 3: 2-ranks decomposition trend vector for case 1.

From Figs. 2 and 3, we can conclude that there was a glith in 59^{th} BPM (BPM Cell08-7) at around July 13, 18:00.

Case 2

For the decompositions of the data collected during 2022-7-13 14:27 ~ 15:25, the *norm_percent*s are shown as Table 3.

Table 3: *Norm_Percent* Values Logged in Case 2

ranks	1	2	3	10
norm_percent	98.98%	99.15%	99.30%	99.38%

In this case, we also investigate the decomposition result for 2-ranks. Similar to the case 1, the 1^{st} rank vectors are almost identical to the 1-rank decomposition vectors. The relative magnitude of 2^{nd} rank to the 1^{st} rank motion in 2-ranks decomposition (λ_2/λ_1 of Eq. (2)) is about 0.6×10^{-3}. That is, in 2-ranks decomposition, the 2^{nd} rank motion can be considered as a weak perturbation.

Figures 4-6 show the 2-ranks decomposition vectors for the case 2. From the figures, we can conclude that the low frequency motion (< 10 Hz) was slightly amplified especially around cell 17 upstream region at 15:15 ~ 20 on July 13.

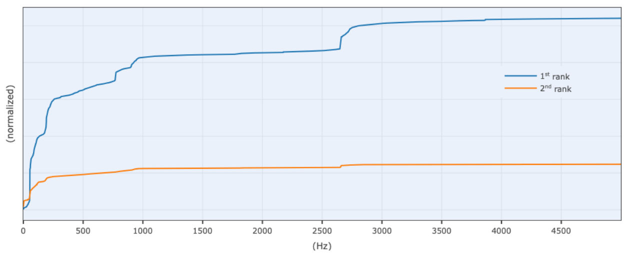

Figure 4: Power spectrum density of 2-ranks decomposition temporal vector for case 2.

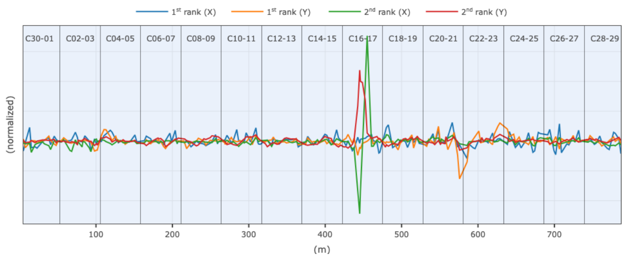

Figure 5: 2-ranks decomposition spatial vector for case 2.

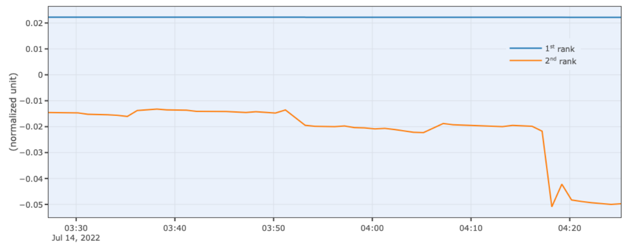

Figure 6: 2-ranks decomposition trend vector for case 2.

SUMMARY

Using the TCA, the NSLS-II storage ring FA data for all BPMs measured every minute are being archived in efficiently compressed files. The decomposition used for the compression are also providing orbit stability information in real time. Also, when the full data are needed, they can be reconstructed from the compressed data so that they are close enough to the original data for any analysis.

REFERENCES

[1] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009. doi:10.1137/07070111X

[2] rTensor, https://cran.r-project.org/web/packages/rTensor