

# THE CORE CONSISTENCY OF A COMPRESSED TENSOR

Georgios Tsitsikas, Evangelos E. Papalexakis

Dept. of Computer Science and Engineering  
University of California, Riverside

## ABSTRACT

Tensor decomposition on big data has attracted significant attention recently. Among the most popular methods is a class of algorithms that leverages compression in order to reduce the size of the tensor and potentially parallelize computations. A fundamental requirement for such methods to work properly is that the low-rank tensor structure is retained upon compression. In lieu of efficient and realistic means of computing and studying the effects of compression on the low rank of a tensor, we study the effects of compression on the core consistency; a widely used heuristic that has been used as a proxy for estimating that low rank. We provide theoretical analysis, where we identify sufficient conditions for the compression such that the core consistency is preserved, and we conduct extensive experiments that validate our analysis. Further, we explore popular compression schemes and how they affect the core consistency.

**Index Terms**— Tensor decomposition, tensor rank, core consistency, CORCONDIA, compression

## 1. INTRODUCTION

In recent years, there has been a tremendous increase in the amount of data being available in many application areas of interest [1]. These data are many times multi-aspect and, therefore, they are very elegantly described by multidimensional arrays where each index of the array corresponds to a specific aspect of the data.

Tensors, which are very closely tied to multidimensional arrays, have proved to be a very useful framework for analyzing and extracting structure from these data, which is usually achieved by employing tensor decompositions. A plethora of such decompositions has been suggested [2, 3, 4], many of which have also seen significant algorithmic advances especially when dealing with big data [5, 6, 7].

A particular scheme of interest, though, is the one where the tensor data are randomly compressed before being decomposed [8, 9, 10]. However, even though the analysis in those papers is sound, all results are predicated on the fact that the rank is preserved during this compression/sketching step. In practical cases, we have observed this to be true, but, to the best of our knowledge, there is no analysis of this phe-

nomenon. Thus, in this paper we set out to investigate the effects of such compression on the rank of a tensor, assuming that it has low-rank trilinear structure.

To this end, and realizing that tensor rank calculation is an NP-hard problem [11], we will focus on the impact of compression on the Core Consistency Diagnostic [12, 13], which is the most widely used heuristic for judging the trilinearity of a tensor and identifying the rank [14, 15, 16]. Specifically, our contributions are:

- **Theoretical analysis:** We provide a sufficient condition for the compression matrices, under which the compressed tensor preserves the Core Consistency.
- **Extensive experimental evaluation:** We thoroughly evaluate our theoretical results using real sugar data that have been chemically verified for their trilinearity [13]. To achieve this, various experiments with different schemes of compression were carried out, and their efficiency in retaining the Core Consistency of a tensor with low-rank trilinear structure is discussed.

## 2. PROBLEM FORMULATION

### 2.1. Notation & Definitions

An  $N$ -mode tensor  $\mathcal{X}$  can be defined as an element of the tensor product of  $N$  vector spaces, and by choosing bases for these spaces, it can be described as a multidimensional array of numbers. Specifically, for the tensor product of  $N$  vector spaces of dimensions  $I_1, I_2, \dots, I_N$ , by choosing bases in  $\mathbb{R}^{I_1}, \mathbb{R}^{I_2}, \dots, \mathbb{R}^{I_N}$ , respectively,  $\mathcal{X}$  can be expressed as an element of  $\mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ .

**Mode- $n$  Fiber:** A mode- $n$  fiber of  $\mathcal{X}$  is the column vector  $\mathcal{X}(\dots, i_{n-1}, :, \dots)$ , whose elements are the elements of  $\mathcal{X}$  for  $i_n = 1, \dots, I_n$  while the rest of the indices are fixed.

**$n$ -Mode Product:** The  $n$ -mode product of  $\mathcal{X}$  with a matrix  $\mathbf{Z} \in \mathbb{R}^{K \times I_n}$ , is denoted by  $\mathcal{X} \times_n \mathbf{Z}$ , and results in a tensor whose  $n$ -mode fibers are the  $n$ -mode fibers of  $\mathcal{X}$  multiplied by  $\mathbf{Z}$ . In other words,

$$(\mathcal{X} \times_n \mathbf{Z})(\dots, i_{n-1}, :, \dots) = \mathbf{Z} \cdot \mathcal{X}(\dots, i_{n-1}, :, \dots)$$

Note that  $(\mathcal{X} \times_n \mathbf{Z}) \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times K \times \dots \times I_N}$ .

**Frobenius Norm:** The Frobenius Norm  $\|\cdot\|$  of a tensor  $\mathcal{X}$  is defined as

$$\|\mathcal{X}\| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} \mathcal{X}(i_1, i_2, \dots, i_N)^2}$$

## 2.2. Tensor Decompositions

Many times we are interested in expressing  $\mathcal{X}$  in a decomposed form, since this can be instrumental in different data analytics scenarios [1, 4].

Particularly, we can decompose  $\mathcal{X}$  as the sum of rank-one tensors. In this paper, we will only consider 3-mode tensors, and, therefore, these rank-one tensors will be the outer product of three vectors,  $\mathbf{a}_p \in \mathbb{R}^I$  with  $p = 1, \dots, P$ ,  $\mathbf{b}_q \in \mathbb{R}^J$  with  $q = 1, \dots, Q$ , and  $\mathbf{c}_r \in \mathbb{R}^K$  with  $r = 1, \dots, R$ . These vectors can be grouped as the columns of three factor matrices  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$ , respectively, so that  $\mathbf{A} \in \mathbb{R}^{I \times P}$ ,  $\mathbf{B} \in \mathbb{R}^{J \times Q}$  and  $\mathbf{C} \in \mathbb{R}^{K \times R}$ .

One such decomposition is PARAFAC [2, 17], which allows us to express  $\mathcal{X}$  as

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \quad (1)$$

or equivalently

$$\mathcal{X} = \mathcal{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \quad (2)$$

where  $R$  is the number of components, and  $\mathcal{I}(i, j, k)$  is 1 for  $i = j = k$  and zero everywhere else.

TUCKER3 is another useful decomposition [3], which generalizes PARAFAC, and allows us to express  $\mathcal{X}$  as

$$\mathcal{X} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R \mathcal{G}(p, q, r) \cdot \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r \quad (3)$$

or equivalently

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \quad (4)$$

where  $\mathcal{G}$  is called the TUCKER3 core.

Additionally, since in our work we assume low-rank structure, we will only consider tall PARAFAC and tall orthonormal TUCKER3 factor matrices. The orthonormality assumption might seem restrictive at first glance, but observe that for non-orthonormal tall TUCKER3 factor matrices we can employ their reduced QR factorization to get

$$\begin{aligned} \mathcal{X} &= \mathcal{G} \times_1 \mathbf{Q}_A \mathbf{R}_A \times_2 \mathbf{Q}_B \mathbf{R}_B \times_3 \mathbf{Q}_C \mathbf{R}_C \\ &= (\mathcal{G} \times_1 \mathbf{R}_A \times_2 \mathbf{R}_B \times_3 \mathbf{R}_C) \times_1 \mathbf{Q}_A \times_2 \mathbf{Q}_B \times_3 \mathbf{Q}_C \\ &= \tilde{\mathcal{G}} \times_1 \mathbf{Q}_A \times_2 \mathbf{Q}_B \times_3 \mathbf{Q}_C \end{aligned}$$

Finally, it should be mentioned that for different values of  $R$  in (1) we get different decompositions, and the same is true for different values of  $P$ ,  $Q$  and  $R$  in (3). Keep in mind, however, that for some values an exact decomposition may not exist at all.

## 2.3. Core Consistency Diagnostic

The Core Consistency Diagnostic (CORCONDIA) [12, 13] is defined as

$$\left(1 - \frac{\|\mathcal{I} - \mathcal{G}\|^2}{\|\mathcal{I}\|^2}\right) \cdot 100$$

where

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{A}^+ \times_2 \mathbf{B}^+ \times_3 \mathbf{C}^+ \quad (5)$$

and  $\mathbf{A}^+$ ,  $\mathbf{B}^+$  and  $\mathbf{C}^+$  are the Moore-Penrose inverses of the PARAFAC factor matrices of  $\mathcal{X}$ .

Expression (5) is obtained as the minimum norm solution of the least squares problem

$$\arg \min_{\mathcal{G}} \|\mathcal{X} - \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}\|$$

Note that PARAFAC can be expressed as the solution of the least squares problem

$$\arg \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \|\mathcal{X} - \mathcal{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}\|$$

Hence, we can see that CORCONDIA essentially attempts to quantify how well a PARAFAC decomposition describes  $\mathcal{X}$ , by comparing to how well  $\mathcal{X}$  can be described when interactions between all the columns of  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$  are allowed.

Specifically, when these interactions do not improve the model significantly, then we can interpret it as a sign that the PARAFAC model is appropriate. Further,  $\mathcal{G}$  will have its dominant elements on the diagonal, and, thus, CORCONDIA will have a value close to 100. On the other hand, when the interactions produce a substantially better model, then the PARAFAC model is probably not appropriate. In fact,  $\mathcal{G}$  will have many off-diagonal elements, which will result in a close to zero, or even negative CORCONDIA.

This means that we can evaluate CORCONDIA on a range of PARAFAC decompositions with different number of components,  $R$ , and select the one with the largest number of components that also retains a reasonably high CORCONDIA value. By using this method, we hope to discover a PARAFAC model that best describes potential trilinear variation in our data.

## 3. PROPOSED ANALYSIS

Although CORCONDIA is a very useful diagnostic for discovering trilinear variation in data, there can be times where it becomes very computationally and memory intensive in practice. Specifically, if we consider very large tensors, not only the PARAFAC factor matrices and their pseudoinverses can take prohibitive amounts of time to calculate, but in cases where the whole tensor cannot fit into the main memory, performance can deteriorate substantially.

For this reason, it would be useful to have in our disposal a tool that allows us to extract and study a much smaller version of the tensor that, despite its small size, retains most

of the systematic variation. However, since finding such a compressed tensor can also be computationally expensive, we need to settle for a trade-off between how fast and how accurately it can be generated.

To this end, we propose to study the statistics of multiple randomly compressed tensors,  $\mathcal{X}'$ , by employing  $n$ -mode products of the uncompressed tensor,  $\mathcal{X}$ , with matrices  $\mathbf{U} \in \mathbb{R}^{L \times I}$ ,  $\mathbf{V} \in \mathbb{R}^{M \times J}$  and  $\mathbf{W} \in \mathbb{R}^{N \times K}$  having orthonormal rows, so that  $\mathcal{X}'$  can be expressed as

$$\mathcal{X}' = \mathcal{X} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W}$$

where  $\mathcal{X}' \in \mathbb{R}^{L \times M \times N}$  with  $L < I$ ,  $M < J$  and  $N < K$ .

The main reason for selecting this kind of compression is on one hand because of its simplicity, but at the same time because it allows us to derive elegant and useful theoretical results, as shown in the following claims.

**Claim 1.** *When  $\mathcal{X}$  has an exact PARAFAC decomposition and its mode-1, mode-2 and mode-3 fibers belong in the rowspace of  $\mathbf{U}$ ,  $\mathbf{V}$  and  $\mathbf{W}$ , respectively, then CORCONDIA is preserved.*

*Proof.* Since  $\mathcal{X}$  has a PARAFAC decomposition, we get

$$\mathcal{X}' = \mathcal{X} \times_1 \mathbf{U} \mathbf{A} \times_2 \mathbf{V} \mathbf{B} \times_3 \mathbf{W} \mathbf{C}$$

and, therefore, the PARAFAC decomposition of  $\mathcal{X}'$  is given by  $\mathbf{A}' = \mathbf{U} \mathbf{A}$ ,  $\mathbf{B}' = \mathbf{V} \mathbf{B}$  and  $\mathbf{C}' = \mathbf{W} \mathbf{C}$ . As a result, (5) gives

$$\begin{aligned} \mathcal{G}' &= \mathcal{X}' \times_1 (\mathbf{U} \mathbf{A})^+ \times_2 (\mathbf{V} \mathbf{B})^+ \times_3 (\mathbf{W} \mathbf{C})^+ \\ &= \mathcal{X} \times_1 (\mathbf{U} \mathbf{A})^+ \mathbf{U} \times_2 (\mathbf{V} \mathbf{B})^+ \mathbf{V} \times_3 (\mathbf{W} \mathbf{C})^+ \mathbf{W} \\ &= \mathcal{X} \times_1 \mathbf{A}^+ \mathbf{U}^T \mathbf{U} \times_2 \mathbf{B}^+ \mathbf{V}^T \mathbf{V} \times_3 \mathbf{C}^+ \mathbf{W}^T \mathbf{W} \\ &= \mathcal{X}_{pr} \times_1 \mathbf{A}^+ \times_2 \mathbf{B}^+ \times_3 \mathbf{C}^+ \end{aligned}$$

where  $\mathcal{X}_{pr} = \mathcal{X} \times_1 \mathbf{U}^T \mathbf{U} \times_2 \mathbf{V}^T \mathbf{V} \times_3 \mathbf{W}^T \mathbf{W}$ , which can be seen as the projection of  $\mathcal{X}$  onto the rowspaces of  $\mathbf{U}$ ,  $\mathbf{V}$  and  $\mathbf{W}$ . Finally, since the fibers of  $\mathcal{X}$  belong in the rowspaces of these matrices, it will hold that  $\mathcal{X}_{pr} = \mathcal{X}$ , which in turn gives

$$\mathcal{G}' = \mathcal{X} \times_1 \mathbf{A}^+ \times_2 \mathbf{B}^+ \times_3 \mathbf{C}^+ = \mathcal{G}$$

Therefore, CORCONDIA is preserved.  $\square$

**Claim 2.** *When  $\mathcal{X}$  has an exact PARAFAC decomposition and we compress using the transpose of its TUCKER3 factor matrices, then CORCONDIA is preserved.*

*Proof.* First, notice that when an exact PARAFAC decomposition exists, then an exact TUCKER3 decomposition will also exist, which can be derived from the PARAFAC decomposition by employing the reduced QR factorization of its factor matrices as shown in subsection 2.2. Now from (4) we get

$$\begin{aligned} \mathcal{X} \times_1 \mathbf{A}^T \times_2 \mathbf{B}^T \times_3 \mathbf{C}^T &= \mathcal{G} \implies \\ \mathcal{X} \times_1 \mathbf{A} \mathbf{A}^T \times_2 \mathbf{B} \mathbf{B}^T \times_3 \mathbf{C} \mathbf{C}^T &= \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \implies \\ \mathcal{X}_{pr} &= \mathcal{X} \end{aligned}$$

and, thus, we conclude that all mode-1, mode-2 and mode-3 fibers belong in the rowspace of  $\mathbf{A}^T$ ,  $\mathbf{B}^T$  and  $\mathbf{C}^T$ , respectively. At this point, all conditions in Claim 1 are satisfied, which means that CORCONDIA is preserved.  $\square$

We should mention that such a compression scheme has also been studied in [18] in the context of speeding up the calculation of PARAFAC decompositions. That said, even though that work can provide further insight into why this compression scheme is sensible, our approach differs in that instead of looking for optimal compression matrices, it takes the more agnostic path of multiple random compressions.

#### 4. EXPERIMENTAL EVALUATION

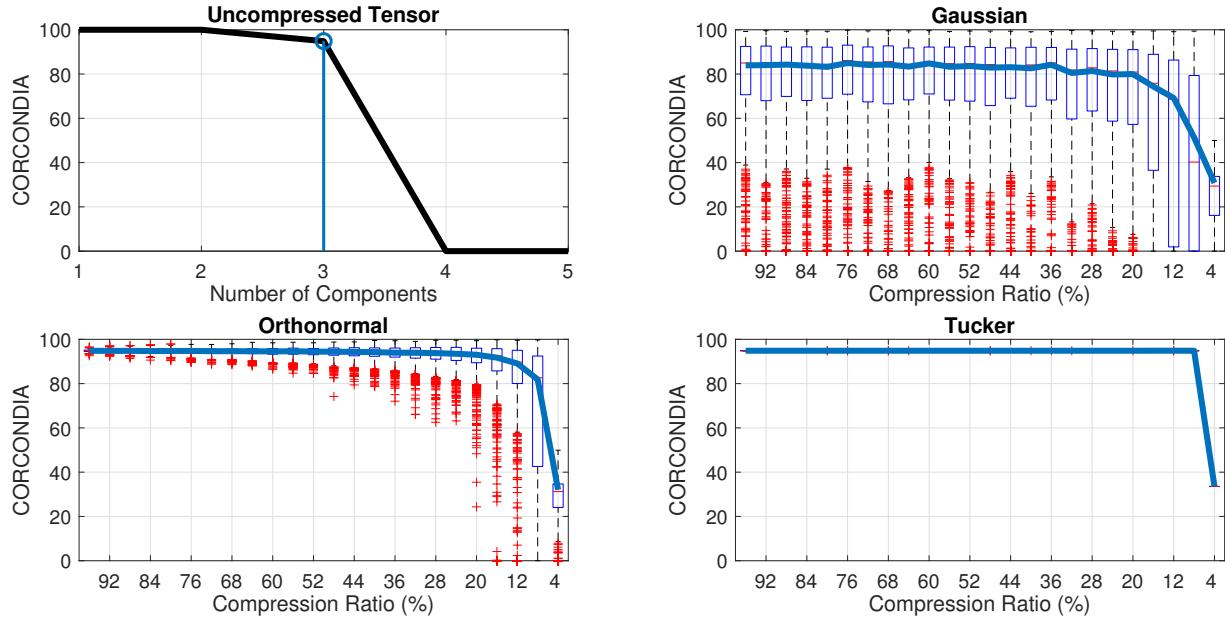
In this section, we present experimental results on the behavior of CORCONDIA on compressed real tensor data. Specifically, we are studying sugar data of size  $268 \times 44 \times 7$  that are known to have trilinear structure [13]. All experiments were run on a system with an Intel(R) Core(TM) i5-7300HQ and 8GB of RAM. Matlab along with Tensor Toolbox [19] was used throughout the whole process, except for the calculation of all PARAFAC and TUCKER3 decompositions for which N-way Toolbox [20] was utilized.

We have experimented with the following three types of compression:

- **Gaussian** - the tensor is multiplied modewise with matrices whose elements are independent identically distributed random variables following the standard normal distribution.
- **Orthonormal** - the tensor is multiplied modewise with random matrices with orthonormal rows. These matrices are obtained from the reduced QR decomposition of the transpose of a Gaussian compression matrix.
- **Tucker** - the tensor is multiplied modewise with the transpose of its TUCKER3 factor matrices, which in fact results in a compressed tensor identical to the TUCKER3 core.

In Figure 1 we present the results for all three types of compression. Specifically, the upper left plot shows the CORCONDIA values of the uncompressed tensor for multiple PARAFAC decompositions with different number of components. The vertical line indicates the fixed number of components at which the compressed tensors are tested, while the circle shows the value of CORCONDIA that they are expected to retain for various compression types and ratios.

The rest of the plots show CORCONDIA for each of the aforementioned compression types, where 1000 samples were used for Gaussian and Orthonormal compression, and 10 samples for Tucker compression. They contain the boxplots for each compression ratio along with the corresponding outliers, which are denoted by '+'. All calculations were done after negative CORCONDIA values were set equal to zero.



**Fig. 1:** Comparison of Gaussian, Orthonormal and Tucker compressions for three PARAFAC components.

Further, the sample means for all compression ratios are represented by the thick lines, and they were calculated after the outliers were first properly smoothed. Finally, the compression ratio is the percentage by which we reduce the size of the first and the second mode; we do not compress in the third mode since it is already really small. For instance, for a compression ratio of 50%, we get a compressed tensor of size  $134 \times 22 \times 7$ . We reserve more detailed compression ratio schemes for the extended version of this work.

Note that CORCONDIA seems to not be preserved only when it starts falling from 100 to 0. This occurs at the PARAFAC model with 3 components, and this is why only this case is discussed in this paper. For 1, 2, 4 and 5 components, it is almost perfectly preserved for all compression types and for all compression ratios up to even 4%.

Examining the plots in Figure 1, it is clear that Tucker compression achieves the best performance by managing to perfectly preserve CORCONDIA up to 8% compression. Orthonormal compression comes second by generally preserving CORCONDIA up to about 20% compression, although in the mean it retains the value high enough up to 8% compression. In fact, we can expect very similar performance even with very few samples since the variance is very small. On the other hand, Gaussian compression has clearly a much worse performance, not only because it rarely retains the original CORCONDIA value, but also because for multiple compressions the variance is too large. That said, it can retain in the mean a high enough CORCONDIA up to 20% compression.

At this point, one might feel tempted to consider Tucker compression as the undisputed winner. However, we should not forget that this compression type requires the calculation

of the TUCKER3 decomposition of the uncompressed tensor which can actually become very computationally expensive. On the other hand, the Gaussian and Orthonormal methods can perform the compression much faster since we can generate the compression matrices easier.

## 5. CONCLUSION

We study the effects of various popular and practical tensor compression schemes in the CORCONDIA of a tensor. We provide theoretical insights on conditions that satisfy perfect retention of CORCONDIA upon compression, along with experimental results that verify our analysis. Further, we evaluate the effect of different random compression schemes to CORCONDIA. Experimental results on real data indicate that it is possible to perform significantly tight compressions without having a serious impact on the value of CORCONDIA. Therefore, this method can be used as a tool to mitigate the consequences of the high time complexity of the calculations that CORCONDIA has to perform, especially on big tensor data.

## 6. ACKNOWLEDGEMENTS

We are grateful to Rasmus Bro and Anne Bech Risum for their valuable feedback, and for providing us with real chemical data to evaluate our compression methods. Research was supported by the National Science Foundation CDS&E Grant no. OAC-1808591 and by the Department of the Navy, Naval Engineering Education Consortium under award no. N00174-17-1-0005. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding parties.

## 7. REFERENCES

- [1] E.E. Papalexakis, C. Faloutsos, and N.D. Sidiropoulos, “Tensors for data mining and data fusion: Models, applications, and scalable algorithms,” *ACM Trans. on Intelligent Systems and Technology*.
- [2] R.A. Harshman, “Foundations of the parafac procedure: Models and conditions for an “explanatory” multimodal factor analysis,” 1970.
- [3] L.R. Tucker, “Some mathematical notes on three-mode factor analysis,” *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.
- [4] Nicholas D Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E Papalexakis, and Christos Faloutsos, “Tensor decomposition for signal processing and machine learning,” *IEEE Signal Processing Magazine*.
- [5] Tamara G Kolda and Jimeng Sun, “Scalable tensor decompositions for multi-aspect data mining,” in *Data Mining, 2008. ICDM’08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 363–372.
- [6] Evangelos E. Papalexakis, Christos Faloutsos, and Nicholas D Sidiropoulos, “Parcube: Sparse parallelizable tensor decompositions,” in *ECML-PKDD’12*.
- [7] U Kang, Evangelos E. Papalexakis, Abhay Harpale, and Christos Faloutsos, “Gigatensor: scaling tensor analysis up by 100 times-algorithms and discoveries,” in *ACM KDD’12*.
- [8] ND Sidiropoulos, Papalexakis, Evangelos E, and C Faloutsos, “A parallel algorithm for big tensor decomposition using randomly compressed cubes (para-comp),” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011.
- [9] N Sidiropoulos, Evangelos E. Papalexakis, and C Faloutsos, “Parallel randomly compressed cubes: A scalable distributed architecture for big tensor decomposition,” *IEEE Signal Processing Magazine*, 2014.
- [10] Bo Yang, Ahmed Zamzam, and Nicholas D Sidiropoulos, “Parasketch: Parallel tensor factorization via sketching,” in *Proceedings of the 2018 SIAM International Conference on Data Mining*. SIAM, 2018, pp. 396–404.
- [11] Johan Håstad, “Tensor rank is np-complete,” *Journal of Algorithms*, vol. 11, no. 4, pp. 644–654, 1990.
- [12] Rasmus Bro, *Multi-way analysis in the food industry: models, algorithms, and applications*, Ph.D. thesis, 1998.
- [13] Rasmus Bro and Henk AL Kiers, “A new efficient method for determining the number of components in parafac models,” *Journal of chemometrics*, vol. 17, no. 5, pp. 274–286, 2003.
- [14] Papalexakis, Evangelos E, “Automatic unsupervised tensor mining with quality assessment,” in *SIAM SDM*, 2016.
- [15] Maja H Kamstrup-Nielsen, Lea G Johnsen, and Rasmus Bro, “Core consistency diagnostic in parafac2,” *Journal of Chemometrics*, vol. 27, no. 5, pp. 99–105, 2013.
- [16] R David Holbrook, James H Yen, and Thomas J Grizzard, “Characterizing natural organic material from the occoquan watershed (northern virginia, us) using fluorescence spectroscopy and parafac,” *Science of the Total Environment*, vol. 361, no. 1-3, pp. 249–266, 2006.
- [17] R. Bro, “Parafac. tutorial and applications,” *Chemometrics and intelligent laboratory systems*, vol. 38, no. 2, pp. 149–171, 1997.
- [18] Rasmus Bro and Claus A Andersson, “Improving the speed of multiway algorithms: Part ii: Compression,” *Chemometrics and intelligent laboratory systems*, vol. 42, no. 1-2, pp. 105–113, 1998.
- [19] B.W. Bader and T.G. Kolda, “Matlab tensor toolbox version 2.2,” *Albuquerque, NM, USA: Sandia National Laboratories*, 2007, available at: <https://www.sandia.gov/~tgkolda/TensorToolbox/> (Oct. 2018).
- [20] C.A. Andersson and R. Bro, “The n-way toolbox for matlab,” *Chemometrics and Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1–4, 2000, available at: <https://www.mathworks.com/matlabcentral/fileexchange/1088-the-n-way-toolbox> (Oct. 2018).