

Compression and noise reduction of field maps

X. Du and L. Groening

GSI Helmholtzzentrum für Schwerionenforschung GmbH, D-64291 Darmstadt, Germany



(Received 9 May 2018; published 23 August 2018)

Errors from discretization and large data volume of field maps is a concern for beam dynamics simulations with respect to achievable accuracy and to the required amount of time. High-order singular value decomposition (HOSVD) has recently emerged as simple, effective, and adaptive tool to extract the essentials from multidimensional data. This paper is on the feasibility of compression and noise reduction of electromagnetic field map data with HOSVD. The method has been applied to an electric field map of a DTL cavity with 11 m in length comprising 55 rf-gaps. The original field map data of 220 MB was converted into practically noise-free data of just 20 KB. Noise was reduced by 95% as demonstrated using a cubic cavity for which the analytical field map is available.

DOI: [10.1103/PhysRevAccelBeams.21.084601](https://doi.org/10.1103/PhysRevAccelBeams.21.084601)

I. INTRODUCTION

The application of finite element methods has increased rapidly. These methods are used frequently to calculate maps of temperatures, material stress, heat flow, and electromagnetic fields. For the design of accelerator components field map data are widely used as well as for plasma physics simulations. Codes for the simulation of particle beam dynamics use electromagnetic field maps calculated previously by external solvers such as CST [1], Super-fish [2], and TOSCA [3]. One major concern in any approximate numerical technique are sources of potential errors in the final solution.

Errors are from spatial discretization, representation of numbers, the solver (as finite number of iterations), and from mesh interpolation. Some of these errors are noise like and others are not. However, there is no strict definition of the term noise and often it is applied intuitively. Numerical errors may be classified as: (1) solution errors: difference between the exact solution of the analytical equations and corresponding numerical solution (2) spatial discretization errors: due to replacing analytical derivatives or integrals of the exact equations by numerical approximations which in turn have truncation errors (3) time discretization errors: time equivalent to spatial discretization errors (4) iteration errors: difference between solution from finite iterations and fully converged solution (5) round-off errors: difference between calculated approximation of a number and its exact mathematical value due to rounding.

Extensive beam dynamics simulation applications, as estimating or optimizing dynamic apertures for instance [4], suffer from large data volume and mesh noise. Field data with many mesh nodes cause large files and are computationally expensive to navigate, while maps with few nodes lack relevant information to perform accurate interpolations, both resulting in decrease of overall accuracy. In the finite element method (FEM), a major source of error is introduced by the spatial discretization of the problem domain into elements, which were studied vastly [5,6], this part of error is defined as noise. For FEM results with tetrahedral meshes, the interpolation to a regular (equidistant) grid is just a technique to mitigate this problem. Adaptive mesh refinement is widely used in FEM to reduce mesh noise in specific regions. However, any algorithm for minimizing such mesh noise in simulation stage is always subject to discretization itself.

Field maps from solvers intrinsically comprise two components. The first inhabits the real physical solution of the specific problem. It is regular, redundant, smooth and differentiable, and it reflects the respective system's symmetries. The other component contains nonphysical simulation errors and discretization noise. Removal of the noise will therefore drastically increase the quality of the data. High order singular value decomposition (HOSVD) is a well established and efficient method for extracting essential information being hidden in high dimensional data such as field maps. The term "high" in HOSVD refers to the order of the tensor to be decomposed. In this paper, we report on successful application of HOSVD to field map data in order to compress and to de-noise the data provided by solvers.

The following section recapitulates the concept of HOSVD following Ref. [7]. The third section introduces the specific case for which an electric field map has been

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

simulated, denoised, and compressed through HOSVD followed by the description of the algorithm and its application to this case. In Sec. V successful benchmarking of the method is presented. Subsection A is on the comparison with an analytical solution (cubic cavity), subsection B is about the evaluation of raw data versus data treated with HOSVD for a DTL cavity, and subsection C investigates the denoised field map with respect to symmetries and boundary conditions, which are satisfied by the analytical solution.

II. INTRODUCTION TO HIGH-ORDER SINGLE VALUE DECOMPOSITION

HOSVD is a multilinear generalization of matrix SVD to high-order tensors, and it can provide for adequate preservation of the data's essentials through representation of a tensor using appropriate bases [7]. This decomposition plays an important role in various domains as spectral analysis [8], communication and radar processing [9], blind source separation [10], and image processing [11,12] for instance. The conceptual basics of HOSVD will be briefly recapitulated following Ref. [7].

A. Definitions

Assume a N th-order $(I_1 \times I_2 \times \dots \times I_N)$ -tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. We refer in general to the column vectors, row vectors etc. of A as its “ n -mode vectors”, where $1 \leq n \leq N$. They are defined as the I_n -dimensional vectors obtained from A by varying the index i_n and keeping the other indices fixed [13], as shown in Fig. 1. The n -mode product of A with a (2×2) -matrix $U^{(n)} \in \mathbb{R}^{J_n \times I_n}$, denoted by $A \times_n U^{(n)}$, is another N th-order tensor of which the elements are given by

$$(A \times_n U^{(n)})_{i_1 i_2 \dots i_{n-1} j_n i_{n+1} \dots i_N} = \sum_{i_n=1}^{I_n} a_{i_1 i_2 \dots i_{n-1} i_n i_{n+1} \dots i_N} u_{j_n i_n}^{(n)}.$$

This operation projects the n -mode vectors of A on the set of base vectors that form $U^{(n)}$.

For simplicity let A and V be ordinary matrices, i.e., 2nd-order tensors, then $A \times_1 U$ is equal to $A \cdot U$ as for ordinary matrix multiplication and

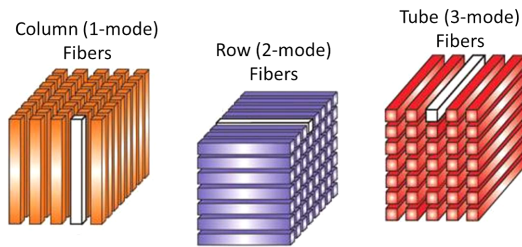


FIG. 1. Different presentations of a tensor (here of 3rd-order). Each fiber represents a n -mode vector, which shall be re-expressed through a new basis.

$$A \times_1 U \times_2 V = V^T \cdot A \cdot U.$$

The n -mode product has the property that for a tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ and the matrices $F \in \mathbb{R}^{J_n \times I_n}$, $G \in \mathbb{R}^{J_m \times I_m}$, the relation

$$(A \times_n F) \times_m G = (A \times_m G) \times_n F = A \times_n F \times_m G \quad (1)$$

is satisfied as shown in [7].

B. n -mode SVD and HOSVD

In the following the matrix (2-order tensor) B is introduced as $B \in \mathbb{R}^{I_n \times (I_1 \cdot I_2 \cdot \dots \cdot I_N / I_n)}$, i.e., as a matrix with I_n columns and $(I_1 \cdot I_2 \cdot \dots \cdot I_N / I_n)$ rows. B shall contain the same information as A just written in another way, namely through a matrix [7]. Each row is made from one n -mode vector and the number of n -mode vectors is equal to the number of rows of B . This non-squared matrix B is decomposed through SVD as

$$B = V^T \cdot N \cdot U^{(n)}, \quad (2)$$

where $N \in \mathbb{R}^{I_n \times I_n}$ is the diagonal matrix from the I_n singular values, $U^{(n)} \in \mathbb{R}^{I_n \times I_n}$ is made from the right-singular vectors of SVD of B , and $V \in \mathbb{R}^{(I_1 \cdot I_2 \cdot \dots \cdot I_N / I_n) \times I_n}$ made from the left-singular vectors. $U^{(n)}$ forms a complete orthonormal base to reexpress all $(I_1 \cdot I_2 \cdot \dots \cdot I_N / I_n)$ n -mode vectors that form B . The required coefficients for this re-expression are included in S_n , such that A is expressed through the new base $U^{(n)}$ as

$$A = S_n \times_n U^{(n)}, \quad (3)$$

thus completing the n -mode SVD of A . An additional m -mode ($m \neq n$) SVD on S_n is performed accordingly through

$$A = S_{mn} \times_m U^{(m)} \times_n U^{(n)}, \quad (4)$$

hence the complete HOSVD transformation of A is

$$A = S_{123 \dots N} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 \dots \times_N U^{(N)}. \quad (5)$$

Equation (1) states that the final result is independent from the order of performing individual i -mode SVDs.

A can be expressed in a more comprehensive way by sorting the base vectors within $U^{(n)}$ according to the magnitude of their respective singular values. In other words, A is expressed through a sum of which the individual summands are sorted according to their relevance such that the first summands are the most relevant ones including the essence of A , while the later summands include more and more irrelevant details or even noise and/or errors. Accordingly, their omission will reduce the data volume and increase the data quality. S is an all-orthogonal and ordered tensor. It is obtained by the inverse HOSVD

from the bases $U^{(n)}$ and it comprises the transformation coefficients.

III. FIELD MAP DATA FROM AN RF-CAVITY

HOSVD is applied to the electric field map of a drift tube linac (DTL) cavity which is currently under design for the upgrade of the UNiversal Linear ACcelerator (UNILAC) at GSI [14]. The map is from CST simulations [1] of the field between drift tubes of the cavity as shown in Fig. 2.

Initially the field data is simulated and stored on an irregular triangular mesh grid. Then the map is projected onto a regular rectangular grid using appropriate interpolation. During the simulations and the projection unavoidable discretization errors occur due to finite resolutions of the mesh grids. Refining the mesh may mitigate discretization errors but it comes at the expense of simulation time. Additionally, it increases the amount of data to be handled thus increasing the efforts to further use it for beam dynamics simulations for instance. Finally, refining may increase the amount of numerical noise and hence even reduce the quality of the field map. In the following it is shown that HOSVD is a very efficient method to separate noise from data essence, hence trimming away errors from discretization directly and to reduce data volume.

The DTL cavity which serves as a test bench for the method comprises 55 cells. To each cell 201*41*41 (long.*hor.*ver.) mesh nodes are assigned. On each node the electric field has three components (E_x , E_y , E_z). The entire field map is stored as a 5th-order tensor and its number of entries along each dimension is (n-cell, z-step, x-step, y-step, spacial component). This tensor is referred to as A in following sections. Accordingly, each scalar field value is uniquely labeled by five numbers, namely: cell, long./hor./ver. node, and spacial component as listed in Table I. The field map data of the whole cavity is stored into

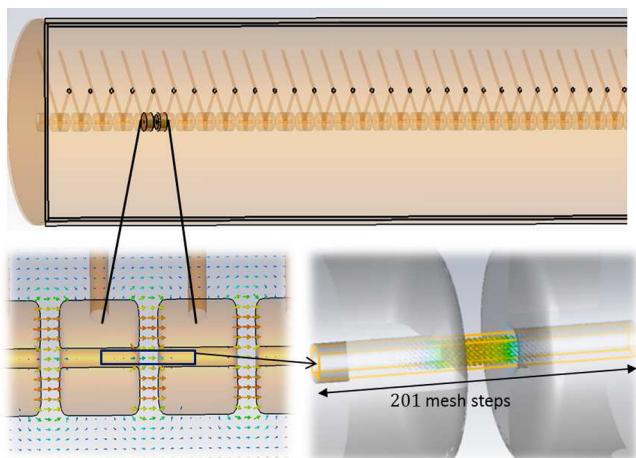


FIG. 2. Electric field map of an Alvarez-type cavity. The cavity diameter is about two meters and the cell length increases from 15 cm to 23 cm. The total cavity length is 11 meters.

TABLE I. Data structure of the electric field map of the DTL cavity.

Label	Dimension of n-mode vector	Maximum number
Cell	I_1	55
Longitudinal node	I_2	201
Horizontal node	I_3	41
Vertical node	I_4	41
Field component	I_5	3

the 5th-order tensor $A \in \mathbb{R}^{55 \times 201 \times 41 \times 41 \times 3}$. The content of A comprises the real physical field map, numerical noise, and discretization errors. As the strictly correct analytical solution for the physical map is not available for cavities including nontrivial drift tube shapes (as it is for a simple box-shaped cavity without drift tubes) the exact amount of noise and errors is not known from first principles. Sufficient convergence of the results with respect to the applied mesh size was assured, revealing randomly distributed effective noise of the results on E_z of about $\pm 1\%$. For $E_{x/y}$ the effective noise increases with the ratio $E_z/E_{x/y}$. Here z , x , y mean the long., hor., ver. direction. Noise is also enhanced at regions with large field gradient $\delta \vec{E} / \delta \vec{r}$ with respect to regions with more homogeneous field.

IV. ALGORITHM FOR COMPRESSION, NOISE REDUCTION, AND RECUPERATION

This section is on the procedure of data compression including the reduction of noise and on the recuperation of denoised data to the same spacial resolution as the original field map data. The procedure is described using the DTL cavity being introduced in the previous section.

A. Compression and noise reduction

Removing noise and redundancy from the original field map through HOSVD is accomplished by several steps. They are defined in the following. (1) The original field map stored in the tensor A is expressed through new bases using the matrices $U^{(n)}$ as

$$A = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \times_4 U^{(4)} \times_5 U^{(5)}, \quad (6)$$

where the so-called core tensor $S \in \mathbb{R}^{55 \times 201 \times 41 \times 41 \times 3}$ includes the projections of A onto the new bases and $U^{(n)} \in \mathbb{R}^{I_n \times I_n}$. Since there is no benefit in representing the three spacial field components through a new basis, $U^{(5)}$ is chosen equal to the (3×3) identity matrix. (2) The contents of the matrices $U^{(n)}$ are reduced by deleting from them the vectors (listed in the rows) that correspond to low singular values. Doing so the dimensions of $U^{(n)}$ are reduced. The cut pendants of $U^{(n)}$ are introduced as $u^{(n)} \in \mathbb{R}^{(C_n < I_n) \times I_n}$ with their amount of rows being reduced to C_n , i.e., the

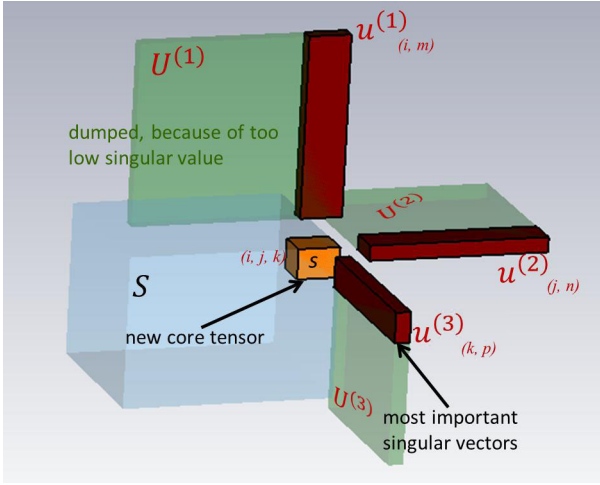


FIG. 3. Illustration of trimming a 3rd-order tensor through SVD. Light green blocks represent the sets of bases $U^{(n)}$ and the blue block represents the initial core tensor S , i.e., the projections of the original data including noise onto these bases. Brown blocks represent the reduced bases $u^{(1)} \in \mathbb{R}^{i \times m}$, $u^{(2)} \in \mathbb{R}^{j \times n}$, $u^{(3)} \in \mathbb{R}^{k \times p}$. The core tensor $S \in \mathbb{R}^{m \times n \times p}$ is reduced to $s \in \mathbb{R}^{i \times j \times k}$ ($i < m, j < n, p < k$) (orange block).

vectors corresponding to the lowest $I_n - C_n$ singular values where removed since they are attributed to noise and/or errors. Accordingly, the corresponding projections comprising the core tensor S need to be deleted as well, hence cutting S to s . This is done by reducing the “thickness” of S along dimension n by removing those layers along n that correspond to the rows being deleted from $U^{(n)}$. This data trimming process is depicted in Fig. 3 using a 3rd-order data volume as an illustrative example. After this noise reduction the remaining denoised data is stored into the tensor \hat{A} expressed through

$$\hat{A} = s \times_1 u^{(1)} \times_2 u^{(2)} \times_3 u^{(3)} \times_4 u^{(4)} \times_5 u^{(5)}. \quad (7)$$

It must be pointed out that both tensors A and \hat{A} are 5th-order tensors $\in \mathbb{R}^{55 \times 201 \times 41 \times 41 \times 3}$, implying that albeit noise has been cut, the effectively available data volume, i.e., its spacial resolution, is preserved. For the example DTL cavity being treated here the thresholds C_n have been chosen such that $s \in \mathbb{R}^{3 \times 8 \times 4 \times 4 \times 3}$, $u^{(1)} \in \mathbb{R}^{3 \times 55}$, $u^{(2)} \in \mathbb{R}^{8 \times 201}$, $u^{(3)} \in \mathbb{R}^{4 \times 41}$, $u^{(4)} \in \mathbb{R}^{4 \times 41}$, and $u^{(5)}$ as the Identity. The choice of these thresholds will be motivated within subsection B. (3) Data compression is through storage of the trimmed data s and $u^{(n)}$ instead of the initial data S and $U^{(n)}$. (4) Recovery of denoised data is done through Eq. (7) and \hat{A} contains the “recuperated from compressed” (RC) data. For further processing of the RC data, field components values f at specific locations are needed. They are retrieved through

$$f_{j_1 j_2 j_3 j_4 j_5} = s \times_1 u_{j_1}^{(1)} \times_2 u_{j_2}^{(2)} \times_3 u_{j_3}^{(3)} \times_4 u_{j_4}^{(4)} \times_5 u_{j_5}^{(5)}, \quad (8)$$

where $f_{j_1 j_2 j_3 j_4 j_5} \in \mathbb{R}$ is the j_5^{th} spacial component of the field on mesh node (j_2, j_3, j_4) within cell number j_1 and $u_{j_n}^{(n)}$ is the j_n^{th} column of $u^{(n)}$.

The compression process has been performed to the map of the DTL cavity. Binary storage of the original data A takes 220 MB of storage volume, while storage of the compressed and denoised data (s and $u^{(n)}$) requires just 20 kB. Accordingly, reduction of storage place by more than four orders of magnitude has been achieved.

The computing process has been coded in Python and decomposition of Eq. (6) takes 16 seconds of CPU time. Additional temporary memory occupation is the same as for the original data volume. Unfolding Eq. (2) enlarges the number of rows of the B matrix (1.3M for 3-mode) and it seems not indicated performing SVD on B due to extremely large memory consumption and CPU time. Therefore n-mode SVD is performed through a square-sum matrix instead of the unfolded matrix B . The square-sum matrix C is calculated dimension by dimension and the example using 3-mode SVD is

$$M_{ijk} = A(1 \sim 55, i, 1 \sim 41, j, k), \quad M_{ijk} \in \mathbb{R}^{55 \times 41}$$

$$C = \sum_{i=1}^{201} \sum_{j=1}^{41} \sum_{k=1}^3 M_{ijk} M_{ijk}^T, \quad C \in \mathbb{R}^{41 \times 41}. \quad (9)$$

Performing SVD on C is much easier than on B giving the same singular vectors if round-off errors are negligible. The corresponding source code is shared on GitHub [15].

In the following section it will be shown that this drastic reduction does not come along with loss of relevant information but that it even increases the quality of the data with respect to its physical content.

B. Threshold strategy

Determining the appropriate thresholds for cutting of $U^{(n)}$ down to $u^{(n)}$ and S down to s is a trade-off between reduction of noise and data volume at one side and of keeping essential information on the field on the other side. Too much cutting causes nonphysical smooth (homogeneous) deviations from the original data. In turn too less cutting will preserve some noise from discretization. Therefore the proper threshold strategy must be established as described below.

Noise is unlikely to repeat itself at different locations of the data. It thus manifests through singular vectors with very low singular values. The physical content of the data, hence its uniformity, is characterized by distributions that occur frequently, by different kinds of symmetries and translation similarity. Singular vectors corresponding to this data essence have high singular values.

The determination is according to the magnitude of the single values corresponding to the vectors forming $U^{(n)}$. For $n = 1, 2, 3$ these singular values have been sorted according to their magnitude and they are partially shown in Fig. 4. The plot for $n = 4$ has been omitted since this case is equivalent to $n = 3$ and for $n = 5$ the plot is obsolete since $U^{(5)}$ has been replaced by the Identity and no reduction is performed. Figure 4 shows that just few singular values stick out with respect to their magnitude from the others. The first singular vector in $U^{(2)}$ is very close to the global mean of E_z along the beam axis. The two largest singular values already differ to each other by about two orders of magnitude. In view of the distribution of singular values the threshold ratio is set to $\sigma = 10^{-6}$ for each set of bases and the size of the preserved (new) core tensor s is accordingly (3,8,4,4,3) as anticipated in the previous subsection.

In order to verify whether these thresholds are indeed properly chosen, they have been varied and the resulting differences $A - \hat{A}$ have been evaluated as illustrated in Fig. 5. The upper part of the figure plots relative differences $E_z(A) - E_z(\hat{A})$ as functions of the x-node and the z-node at DTL cell #44 at y-node #25. These plots are shown for different thresholds being applied along the first dimension of A (see Table I) starting from keeping (3,8,4,4,3) singular values along each of the five dimensions. This example illustrates that the plots do essentially not change if more than three singular values are kept and that the relative differences are below 1.0%. This observation is in full agreement to the upper part of Fig. 4, which shows that just three singular values stick out notably from the others, i.e., the field map essence along dimension #1 is included in just three singular values out of 55. The lower part of Fig. 5

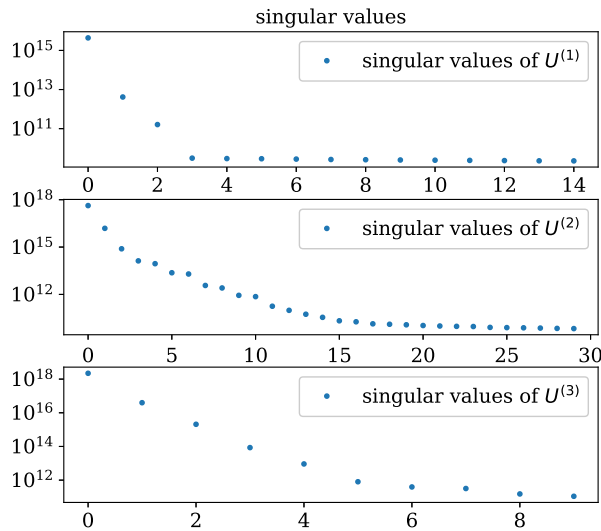


FIG. 4. Highest singular values for $U^{(1)}$, $U^{(2)}$ and $U^{(3)}$ for the field map of the DTL cavity. They are sorted according to their magnitude being extracted through HOSVD.

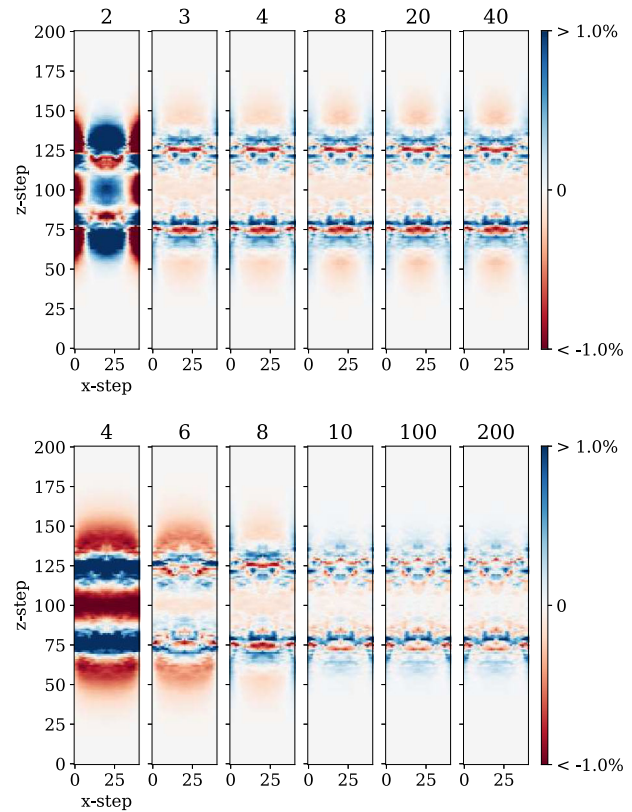


FIG. 5. Relative differences of E_z from original A and “recuperated from compressed” field map \hat{A} as a function of the x-node (abscissa) and the z-node (ordinate) at DTL cell #44 at y-node #25. Differences are shown for different amounts of singular values being preserved (upper label) in the 1st/2nd (upper/lower figure part) dimension of A .

shows the corresponding results for varying the threshold along the second dimension of A . The field differences do not change with the amount of kept singular values if this amount is chosen larger than eight. Also this finding is in

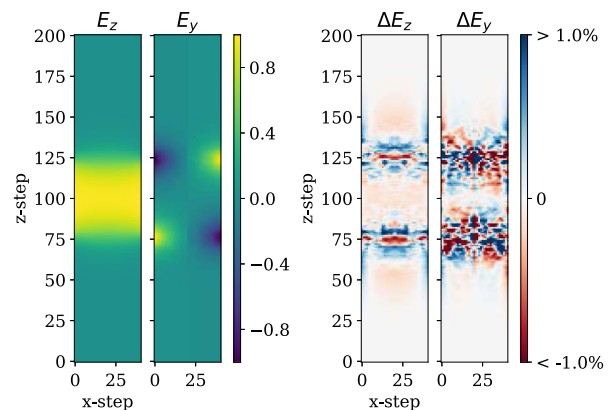


FIG. 6. Left: E_z and E_y extracted from \hat{A} as functions of the x-node (abscissa) and the z-node (ordinate) at DTL cell #44 at y-node #25. Right: relative differences $E_z(A) - E_z(\hat{A})$ and $E_y(A) - E_y(\hat{A})$ as functions of the x-node (abscissa) and the z-node (ordinate) at DTL cell #44 at y-node #25.

agreement with the centre part of Fig. 4 indicating that the field map essence along dimension $\#2$ is included in just eight singular values out of 201. Compared to transformations that use fixed bases as discrete cosines and wavelets, HOSVD bases are created from the specific multidimensional data and hence they are more adapted to the intrinsic structures of this data. This adaptivity enables HOSVD being a compressing transformation for field map data inhabiting considerable uniformity. The section is closed by showing that noise indeed is highest at regions with enhanced field inhomogeneity, i.e., with large $\delta\vec{E}/\delta\vec{r}$ as already mentioned in Sec. III. Figure 6 shows the electric field components of cell $\#44$ at y-node $\#25$ and keeping (3,8,4,4,3) singular values. Close to the drift tubes the transverse field components are largest as well as the difference of the vertical field components $E_y(A) - E_y(\hat{A})$.

V. BENCHMARKING OF THE HOSVD METHOD

At best case HOSVD-based processing provides field maps that exactly correspond to analytical solutions especially for scenarios where these solution cannot be found as for gaps formed by drift tubes inhabiting nontrivial geometries for instance [16]. Accordingly, the difference between A and \hat{A} should be just from noise and errors. The HOSVD-based method was benchmarked by comparing its results to an analytical solution using the well-understood cubic cavity depicted in Fig. 7. Subsection A is on the comparison of the results from HOSVD to the analytical solution while subsection B is on the smoothness of the de-noised field map obtained from \hat{A} , i.e., from HOSVD.

A. Comparisons to analytical solution

The analytic expressions for the electric field components inside a cubic cavity are

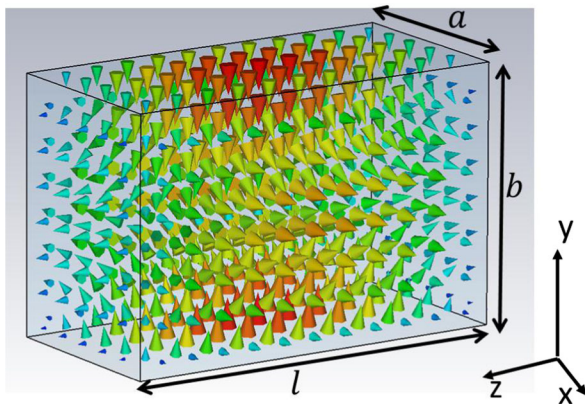


FIG. 7. Simulated electric field map of the TM_{111} -mode inside a cubic cavity. Colors correspond to absolute field values.

$$\begin{aligned} E_x &= E_1 \cos\left(\frac{m\pi}{a}x\right) \sin\left(\frac{n\pi}{b}y\right) \sin\left(\frac{p\pi}{l}z\right), \\ E_y &= E_2 \sin\left(\frac{m\pi}{a}x\right) \cos\left(\frac{n\pi}{b}y\right) \sin\left(\frac{p\pi}{l}z\right), \\ E_z &= E_3 \sin\left(\frac{m\pi}{a}x\right) \sin\left(\frac{n\pi}{b}y\right) \cos\left(\frac{p\pi}{l}z\right), \\ E_1m + E_2n + E_3p &= 0, \end{aligned} \quad (10)$$

where a, b, l are the width, height, and length of the cavity, and m, n, p are the corresponding wave numbers. The analytical field has been scaled to match the amplitude of the simulated field map. In order to approximately match the size of a DTL cell, the dimensions of the cubic cavity were chosen as

$$\begin{aligned} a &= 100 \text{ mm}, \quad b = 130 \text{ mm}, \quad l = 200 \text{ mm}, \\ m &= 1, n = 1, p = 1 \end{aligned} \quad (11)$$

and the cavity has been simulated using CST-MWS thus providing the discrete numerical field map shown in Fig. 7.

The simulations were done using a fine triangular mesh (max. mesh steps per wave length = 40, 2nd-order solver) and using a rough mesh (max. mesh step per wave length = 20, 1st-order solver), respectively. The simulated field map is exported as 4th-order tensor $A \in \mathbb{R}^{100 \times 50 \times 50 \times 3}$. Field maps from both simulations were denoised applying the threshold strategy described in Sec. IV. The numbers of preserved singular values are (2,2,2,3).

The original noisy field map and the RC field data have been compared to the analytical field. Figure 8 depicts these

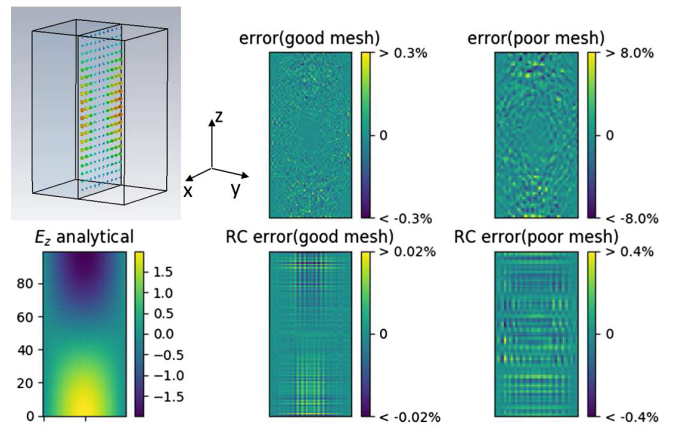


FIG. 8. Upper left: definition of the cutting plane within the cubic cavity. Lower left: analytic E_z at the cutting plane. Upper center: relative difference of E_z from analytical solution and from simulations with fine mesh. Upper right: relative difference of E_z from analytical solution and from simulations with rough mesh. Lower center: relative difference of E_z from analytical solution and from HOSVD starting from fine mesh simulation. Lower right: relative difference of E_z from analytical solution and from HOSVD starting from rough mesh simulations.

comparisons revealing that the algorithm reduces significantly the amount of noise for both mesh types and reproduces the analytical field values very well. Using fine mesh simulations, transverse field values differ less than 1% from the analytical field values, being sufficiently accurate for most beam dynamics calculations. However, such fine meshes are not practical for realistic cavity geometries that are much larger in volume and inhabit more complex local shapes. In order to simulate the latter adequately, the respective local mesh must be refined at expense of the mesh resolution elsewhere. Hence, limitations in memory and in reasonable CPU time require effective reduction of mesh resolution down to values equivalent to the rough mesh scenario treated here. Simulations using the rough mesh reveal deviations of the transverse field with respect to exact solution of almost 10%, being definitely too large for reliable beam dynamics simulations.

However, Figure 8 demonstrates that HOSVD is a method that can extract the essence of field map data, i.e., field values deviating from the analytical solution by less than 1% even from poor-meshed simulations inhabiting large errors. Additionally, the data volume to be stored is significantly reduced during this extraction.

In order to estimate the error that might be introduced by the method, the compression and denoise algorithm is applied to a field map known from the precise analytical solution. To this purpose the analytical solution is projected onto mesh nodes. For such noiseless data, the output of the algorithm is supposed introduce no error for the adequate threshold strategy. The upper part of Fig. 9 plots the singular values for the analytical field, showing that already the third and following singular values drop to 10^{-14} , which can be treated as zeros numerically. The lower part shows the deviation of the RC analytical field from the original

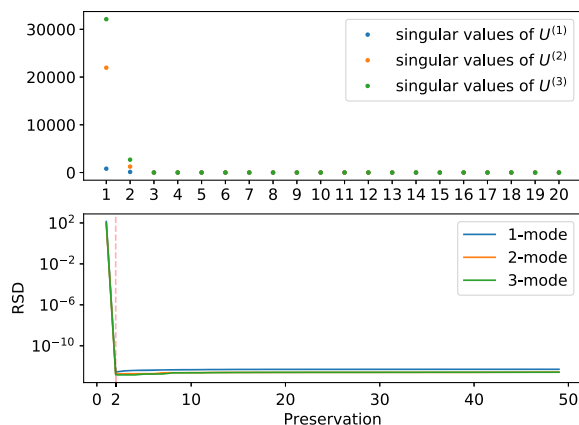


FIG. 9. Upper: sorted singular values obtained from HOSVD of the analytical field map of a cubic cavity represented by Eq. (10) and projected onto a mesh. Lower: RC analytical and original analytical RSD (root-square deviation) as functions of preservation for different mode.

analytical field, demonstrating that preserving the 1st and 2nd singular values of each mode is sufficient to represent entire analytical field. Hence, the algorithm does not introduce additional errors from unphysical smoothing for instance, that outweigh its advantages from compression and denoising.

B. Comparison of smoothness

Analytical solutions of fields within vacuum are intrinsically smooth, i.e., they fulfill the Laplace equation. Additionally they reflect symmetries of surrounding geometries. However, maps that visualize the field vectors do not allow for resolving the field smoothness since the fluctuations are still too tiny to be resolved at a glance. To this end field derivatives $\delta E_i / \delta r_j$ are much more appropriate. The smoothness of field derivatives from simulated data and from HOSVD-processed simulations has been evaluated and compared. Figure 10 shows the derivatives of E_z and E_y along a DTL cell at an off-axis distance of 5 mm. While the originally simulated field shows non-physical fluctuations violating the Laplace equation, the field from HOSVD-trimming is perfectly smooth and reflects the scenarios' symmetries. Accordingly, HOSVD-trimming can provide for field maps that are very close to analytical solutions even for very complex geometries and large volumes.

C. Symmetries

To evaluate the field map quality through identifying physically imposed symmetries, local field distributions of the 55th cell are taken from different evaluation paths which are supposed to have exactly the same analytical solution because of axial symmetry in the x-y plane and mirror symmetry with respect to the z-axis. Eight different paths with distance of ten steps from the beam axis are used for this evaluation (six steps away on one direction

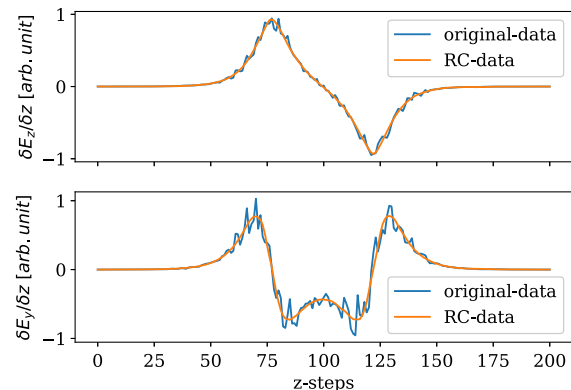


FIG. 10. Comparison of the derivatives $\delta E_z / \delta z$ (upper) and $\delta E_y / \delta z$ (lower) along the z-axis along a DTL cell for simulated data and for simulated data being processed through HOSVD. The latter is much smoother and features the symmetries imposed by the geometry.

and eight steps away on another direction, makes $6^2 + 8^2 = 10^2$). They have identical lengths and extend from the entrance/exit of the drift tube to the gap center, the latter being the symmetry plane of the cell. As reference the field distribution along the beam axis of the 55th cell is extracted as $f(55, 0 \sim 201, 21, 21, 3)$, where the function f is defined by Eq. (8). These eight field distributions are evaluated as

$$\begin{aligned}
 E_z(1) &= f(55, 41 \sim 101, 27, 29, 3) \\
 E_z(2) &= f(55, 41 \sim 101, 29, 27, 3) \\
 E_z(3) &= f(55, 41 \sim 101, 13, 15, 3) \\
 E_z(4) &= f(55, 41 \sim 101, 15, 13, 3) \\
 E_z(5) &= f(55, 161 \sim 101, 27, 29, 3) \\
 E_z(6) &= f(55, 161 \sim 101, 29, 27, 3) \\
 E_z(7) &= f(55, 161 \sim 101, 13, 15, 3) \\
 E_z(8) &= f(55, 161 \sim 101, 15, 13, 3), \quad (12)
 \end{aligned}$$

where $41 \sim 101$ refers to the low energy half of the cell and $101 \sim 161$ refers to the high energy end of the cell. Both, de-noised data and original data, are sampled with Eq. (12) and compared to each other. The original field is sampled by the same process and it is compared with the de-noised data as shown in Fig. 11. The amount of individual field deviations for each path with respect to the average is reduced by the algorithm by the factor of four. Although noise has been suppressed, persisting simulation errors (see Introduction) were not eliminated. The denoise algorithm reduces randomly distributed noise, but some low frequency contents of the error included in the data remains

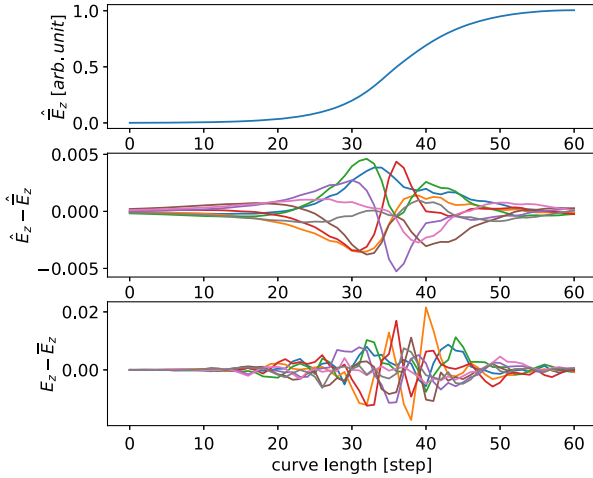


FIG. 11. Deviation of sampled field distributions $E_z(1 \sim 8)$ from their average. Upper: axial field along the half of the 55th DTL cell obtained from averaging the eight paths. Center: individual differences from the eight paths to the average for the de-noised data. Lower: individual differences from the eight paths to the average for the original data.

since it is not identified as noise but rather as simulation error. For an analogy can be stated that compression of data files containing music suppress the amount of noise but it cannot improve the quality of the composition itself. Integrated dipole components are another criterion for quantifying the quality of field maps. The field map of the 55th DTL cell is integrated along the z-axis through

$$\begin{aligned}
 G_x(i, j) &= \sum_1^{201} f(55, 1 \sim 201, i, j, 1) \simeq 0 \\
 G_y(i, j) &= \sum_1^{201} f(55, 1 \sim 201, i, j, 2) \simeq 0 \\
 G_z(i, j) &= \sum_1^{201} f(55, 1 \sim 201, i, j, 3) \simeq \text{constant}. \quad (13)
 \end{aligned}$$

G_x and G_y , being the integrated dipole field components inside the cell, must be zero since transverse fields symmetrically cancel along the full cell length. G_z should be constant as the integral always starts at the drift tube center (equipotential area) and end at the subsequent cell center.

Figure 12 shows that G_x , G_y , and G_z (2D maps of 41×41 pixels) barely differ for denoised data and original data, indicating that the algorithm does not change these integrals being global statistical properties of the field map. Integration of random noise is supposed to be of negligible amount. The total on axis voltage G_z and the traverse kicks G_x and G_y are preserved as data essence, even if they inhabit some deviation from the analytical solution. This is because boundary conditions are not used by the algorithm, and hence non-noise like errors in the compressed data cannot be mitigated.

One method for using symmetries to further improve the field map quality is to use a coordinate system that inhabits the systems intrinsic symmetries. For the map treated here, cylindrical coordinates (n, z, r, θ, c, b) are suitable, where n is the cell number ($1 \sim 55$), z is the number of z-steps ($1 \sim 201$), r is the number of radial steps ($1 \sim 20$), θ is the

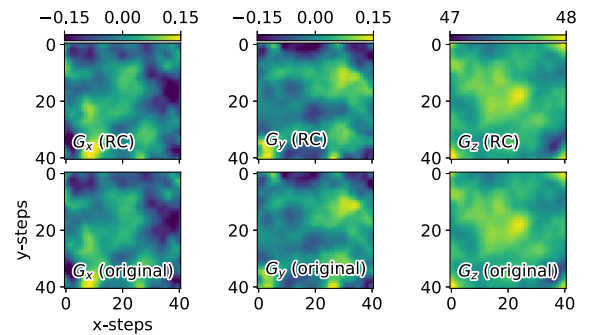


FIG. 12. Transverse (G_x , G_y) and longitudinal field integral G_z calculated from Eq. (13) for denoised data (RC) and from original data.

number of angular steps ($1 \sim 45$), c is the spacial field components ($1 \sim 3$), and b is the two element vector ($1 \sim 2$) containing the field value on both sides of the gap center. After HOSVD, the cylindrical symmetry is established by reducing the row number of $u^{(a)}$ from 45 to 1 and the mirror symmetry is established by reducing the row number of $u^{(b)}$ from 2 to 1 as well. Such a process modifies the field map following physics reasoning instead of aiming at denoising. However, this approach is beyond the scope of this paper.

VI. CONCLUSION

Efficient denoising and hence compression of high dimensional electric field map data by HOSVD has been proposed and successfully applied. The electric field map of an DTL cavity comprising 55 rf-gaps obtained from CST simulations was compressed from an initial data volume of 220 MB by four order of magnitude to just 20 kB. Noise has been practically removed from the data and the data quality increased significantly with respect to redundancy, smoothness, and symmetry. The extraction of denoised data is done through multilinear mapping rather than through time consuming memory search. This method is general and applicable to all kind of maps. It will significantly increase the accuracy and hence reliability of codes that apply such maps.

-
- [1] <https://www.cst.com/Applications/MWandRF>.
 - [2] http://laacg.lanl.gov/laacg/services/sfu_04_04_03.phtml.
 - [3] <https://www.3ds.com/products-services/simulia/products/tosca/>.
 - [4] Weiwei Gao, Lin Wang, and Weimin Li, Simultaneous optimization of beam emittance and dynamic aperture for electron storage ring using genetic algorithm, *Phys. Rev. ST Accel. Beams* **14**, 094001 (2011).
 - [5] H. Schmidt, T. Alber, T. Wehner, R. Blakytyn, and H. J. Wilke, Discretization error when using finite element

models: analysis and evaluation of an underestimated problem, *J. Biomech.* **42**, 1926 (2009).

- [6] R. Lee and A. C. Cangellaris, A study of discretization error in the finite element approximation of wave solutions, *IEEE Trans. Antennas Propag.* **40**, 542 (1992).
- [7] L. De Lathauwer, B. De Moor, and J. Vandewalle, A multilinear singular value decomposition, *SIAM J. Matrix Anal. Appl.* **21**, 1253 (2000).
- [8] N. Kreimer and M. D. Sacchi, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation, *Geophysics* **77**, V113 (2012).
- [9] B. Song, F. Roemer, and M. Haardt, Blind estimation of SIMO channels using a tensor-based subspace method, in *Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers* (IEEE, Piscataway, NJ, 2010), pp. 8–12.
- [10] V. Zarzoso, A. K. Nandi, and E. Bacharakis, Maternal and foetal ECG separation using blind source separation methods, *IMA Journal of mathematics applied in medicine and biology* **14**, 207 (1997).
- [11] X. Zhang, J. Peng, M. Xu, W. Yang, Z. Zhang, H. Guo, W. Chen, Q. Feng, E. X. Wu, and Y. Feng, Denoise diffusion-weighted images using higher-order singular value decomposition, *NeuroImage* **156**, 128 (2017).
- [12] A. Rajwade, A. Rangarajan, and A. Banerjee, Image denoising using the higher order singular value decomposition, *IEEE Trans. Pattern Anal. Mach. Intell.* **35**, 849 (2013).
- [13] T. G. Kolda and B. W. Bader, Tensor decompositions and applications, *SIAM Rev.* **51**, 455 (2009).
- [14] L. Groening and S. Mickat, Development of new Heavy Ion Linacs at GSI, in *Proceedings of the XXVIII Linear Accelerator Conference, East Lansing, MI, USA*, edited by Y. Yamazaki (MSU, East Lansing, 2016).
- [15] <https://github.com/duxngsi/Field-map-HOSVD>.
- [16] X. Du, A. Seibel, S. Mickat, and L. Groening, Alvarez DTL cavity design for the UNILAC Upgrade, in *Proceedings of the 6th International Particle Accelerator Conference, Richmond, VA, USA*, edited by S. Henderson (ANL, Richmond, 2015).