



**HAL**  
open science

# Fast BEM solution for scattering problems using Quantized Tensor Train format

Jean-René Poirier, Ayoub Bellouch, Olivier Coulaud, Oguz Kaya

► **To cite this version:**

Jean-René Poirier, Ayoub Bellouch, Olivier Coulaud, Oguz Kaya. Fast BEM solution for scattering problems using Quantized Tensor Train format. COMPUMAG 2019 - 22nd International Conference on the Computation of Electromagnetic Field, Jul 2019, Paris, France. hal-02264277

**HAL Id: hal-02264277**

**<https://hal.science/hal-02264277>**

Submitted on 19 Sep 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Fast BEM solution for scattering problems using Quantized Tensor-Train format

J.-R. Poirier<sup>1,2</sup>, A. Bellouch<sup>2</sup>, O. Coulaud<sup>2</sup>, and O. Kaya<sup>3</sup>

<sup>1</sup>LAPLACE, Université de Toulouse, CNRS, INPT, UPS, Toulouse, France

<sup>2</sup>INRIA Bordeaux Sud-Ouest-HIEPACS

<sup>3</sup>Université Paris Saclay

It is common to accelerate the boundary element method by compression methods (FMM, H-Matrix / ACA) that enable a more accurate solution or a solution in higher frequency. In this work, we present a compression method based on a transformation of the linear system into Tensor-Train format by the quantization technique. The method is applied to a scattering problem by a canonical object with a regular mesh and improves the performance obtained by the previous methods.

*Index Terms*—Boundary integral equations, electromagnetic diffraction, Quantized Tensor-Train

## I. INTRODUCTION

**M**ANY applications in science and engineering are formulated in terms of boundary integral equations. This is namely the case in electromagnetic scattering to avoid the use of an artificial condition to truncate the domain of study.

From a numerical point of view, the main difficulty is the solution of a full complex linear system

$$Ax = b.$$

Many compression and solution accelerations of this linear system have been developed and allowed to increase the maximum accessible sizes from a few thousand to several millions. Continuing to improve performance and accelerate solution remains a topical issue. In this work we propose the use of a tensor format for solving the algebraic problem.

As numerical example, we will consider the scattering by a perfectly conducting cylinder  $\Gamma$  in the E-polarization ( $u = E_z$ ). The involved boundary-value problem to solve is then the Helmholtz equation with a Dirichlet boundary condition and a radiation condition at infinity.

The BEM solution can be written with a single layer potential [1],

$$\int_{\Gamma} G(x, x_s) j(x_s) d\gamma(x_s) = -u^{\text{inc}}(x), \forall x \text{ on } \Gamma, \quad (1)$$

where  $\Gamma$  is the boundary,  $u^{\text{inc}}$  the incident electric field,  $j$  the sought density current and  $G$  the Green's function.

In free space, this Green function is usually given by  $G(x, x_s) = \frac{1}{4i} H_0^{(2)}(k|x - x_s|)$  where  $i$  is the imaginary unit,  $k$  the wave number and  $H_0^{(2)}$  the Hankel function of second kind.

## II. SOLUTION WITH TENSOR TECHNIQUES

For many years various types of problems have been formulated using tensors instead of a classical linear matrix algebra [3]. More recently, tools have been developed to treat high order tensors (dimension higher than 3) based on the implementation of low rank techniques to effectively reproduce the algebraic structure of the system. In the literature, there are two main formats in terms of tensors for an efficient and stable

”hollow” representation of a very large system. These are the approximations by Tensor-Train (TT) and Hierarchical Tucker decompositions. In this work we have chosen to apply the TT format to formulate the integral equation problem for a fast solution.

### A. Tensor-Train Format

The TT format [2] consists of writing a  $d$ -dimensional tensor  $\mathcal{T} \in \mathbb{R}^{n_1 \times \dots \times n_d}$  as a chain of 3-dimensional tensors according to the formula

$$\mathcal{T}(i_1, \dots, i_d) = \sum_{\alpha_0, \dots, \alpha_d} G_1(\alpha_0, i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \dots G_d(\alpha_{d-1}, i_d, \alpha_d)$$

which can be written in a compact form

$$\mathcal{T}(i_1, \dots, i_d) = \underbrace{G_1[i_1]}_{r_0 \times r_1} \underbrace{G_2[i_2]}_{r_1 \times r_2} \dots \underbrace{G_d[i_d]}_{r_{d-1} \times r_d}$$

where

- $G_i$ : TT-cores (matrices of size  $r_{k-1} \times r_k$  with  $r_0 = r_d = 1$ )
- $r_i$ : TT - rank
- $r = \max(r_i)$ ,  $r$  maximal rank of cores.

The complexity of the storage is  $O(dnr^2)$  for  $O(n^d)$  elements. If  $r$  is small the tensor is of low rank and the storage and as a result the solver will be fast.

The same format is given for a linear operator in  $\mathbb{R}^{n_1 \times \dots \times n_d}$  which is represented by a  $2d$ -dimensional tensor  $\mathcal{A}$  that couple elements as  $(i_n, j_n)$  for  $n = 1 \dots d$  resulting from the couple (i,j). TT representation of this tensor has the compact form

$$\mathcal{A}(i_1, \dots, i_d; j_1, \dots, j_d) = \underbrace{M_1[i_1, j_1]}_{r_1 \times r_1} \underbrace{M_2[i_2, j_2]}_{r_2 \times r_2} \dots \underbrace{M_d[i_d, j_d]}_{r_d \times r_d}$$

where  $M_k[i_k, j_k]$  is a matrix of rank  $r_{k-1} \times r_k$ .

In TT format, we have the corresponding algebraic operations (addition, matrix-vector product, dot product, etc.) that are fast because of the reduced storage of the tensor.

## B. Quantization and QTT format

Although some physical problems can naturally be formulated in terms of tensors and therefore solved by adapted techniques, a key point for the application of a tensor compression technique on integral equations is quantization [6], which enables a transition from a vector or a matrix to a tensor representation. For a vector  $x \in \mathbb{R}^I$  and  $I, d \in \mathbb{N}$  such that

$$I = I_1 I_2 \dots I_d \text{ where } I_k \in \mathbb{N} \text{ and for } k = 1, \dots, d$$

we can transform  $x$  to a  $d$ -dimensional tensor  $\mathcal{X}$  as

$$x_i = \mathcal{X}(i_1, \dots, i_d) \text{ with } i_k = 0, 1, \dots, I_k - 1 \text{ for } k = 1, \dots, d$$

using the following *flattening* of the multi-index  $(i_1, \dots, i_d)$ :

$$i = i_1 + i_2 I_1 + \dots + i_d I_1 I_2 \dots I_{d-1}.$$

An equivalent procedure can be applied to obtain a QTT representation for  $A \in \mathbb{R}^{I \times I}$  with quantization in both dimensions.

## III. SOLVING BOUNDARY ELEMENT METHOD WITH TENSOR-TRAIN FORMAT

### A. Compression with TT-cross algorithm

A tensor  $\mathcal{X}$  in full format can be approximated by  $\tilde{\mathcal{X}}$  in the low-rank Tensor-Train (TT) format with the TT-SVD algorithm [5] such that

$$\|\tilde{\mathcal{X}} - \mathcal{X}\|_F \leq \epsilon \|\mathcal{X}\|_F$$

where  $\|\mathcal{X}\|_F$  is the Frobenius norm and  $\epsilon$  is the desired accuracy.

This procedure being too expensive, it is convenient to use a TT-cross algorithm with TT-rounding [5] that is a generalization of the well-known cross algorithm used for matrices. It build

$$\tilde{\mathcal{X}} = f(\tilde{\mathcal{A}}, \tilde{\mathcal{B}}, \dots)$$

where  $\tilde{\mathcal{A}}, \tilde{\mathcal{B}}$  are cross approximations of cores  $\mathcal{A}, \mathcal{B}$  from the TT-SVD algorithm.

The numerical implementation is done with the use of the Python TT-Toolbox from Oseledets. The considered numerical example here is a metallic cylinder of radius 1m illuminated by an electromagnetic wave at frequency 0.6GHz. It is described with a regular mesh with  $N = 2^d$  elements. Table I gives the memory storage required by the matrix  $A$ .

d	N	Full Matrix	TT-cross	TT-rounding
6	64	0.0625 Mb	0.067 Mb	0.007 Mb
10	1024	16 Mb	0.33 Mb	0.048 Mb
14	16384	4 Gb	0.54 Mb	0.063 Mb
16	65536	64 Gb	0.7 Mb	0.073 Mb
18	262144	1 Tb	0.75 Mb	0.079 Mb
20	1048576	16 Tb	1 Mb	0.084 Mb

TABLE I  
MEMORY STORAGE FOR THE MATRIX A

These results show clearly a very good compression arising from this method even for very small value of  $N$ .

## B. Solution using AMEN solver

The solution of the linear system may be performed by a Krylov method applying the fast vector matrix product. In this work we had applied the AMEN method developed specifically for this tensor format [7]. The results are given Table II for  $\epsilon = 0.001$ .

d	N	full solution $x_{ref}$	Amen $x_{TT}$	Error $\ x_{TT} - x_{ref}\ $
6	64		0.66	1.6E-11
8	256		1.23	1.09E-3
10	1024		2.15	1.78E-3
12	4096		3.57	4.5E-3
14	16384		5.41	
16	65536			
18	262144			
20	1048576			

TABLE II  
COMPUTATION TIMES (IN SECONDS) AND ERROR WITH REFERENCE SOLUTION

As expected the solver make it possible to obtain very good performance due to the reduced storage and while keeping the desired accuracy.

## C. Comparison with the H-Matrix techniques

We give in Table III the compression obtained on the same example with H-matrix format using an ACA technique. As

N	QTT/TT-Rounding	H-Matrix
64	10.76 %	100 %
256	2.88 %	37.25%
1024	0.28 %	13.6%
4096	0.0208%	4.34%
16384	$1.4 \cdot 10^{-3}$ %	1.2%
65536	$1.06 \cdot 10^{-4}$ %	0.30 %

TABLE III  
COMPARISON OF COMPRESSION RATE  $100 \frac{\text{size}(A_{TT})}{\text{size}(A_{Full})}$  WITH H-MATRIX

a result compression and performance of the tensor solution are much higher than those obtained by H-matrix techniques. The QTT technique is known to be efficient for regular meshes and solution. The performance on more complex problems or geometries can be less efficient [4] and this tensor technique will be used as an efficient preconditioner in those cases.

## REFERENCES

- [1] R. F. Harrington, "Field Compilation by Moment Methods" *Macmillan*, New York, 1968.
- [2] I. V. Oseledets. 2011. "Tensor-Train Decomposition" *SIAM J. Sci. Comput.*, vol 33, no 5 September 2011, pp 2295-2317.
- [3] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [4] Eduardo Corona, Abtin Rahimian, Denis Zorin, "A Tensor-Train accelerated solver for integral equations in complex geometries" *Journal of Computational Physics* Volume 334, April 2017, Pages 145–169
- [5] Ivan Oseledets, Eugene Tyrtyshnikov, "TT-cross approximation for multidimensional arrays" *Linear Algebra and its Applications*, Volume 432, Issue 1, 2010, Pages 70-88.
- [6] Khoromskij, B. "O (d log N)-Quantics Approximation of N - d Tensors in High-Dimensional Numerical Modeling" *Constructive Approximation*, vol 34, 2009.
- [7] Sergey V. Dolgov and Dmitry V. Savostyanov. "Alternating Minimal Energy Methods for Linear Systems in Higher Dimensions". *SIAM Journal on Scientific Computing* vol 36, no 5, pp. A2248-A2271, 2014.