Universität des Saarlandes

Fachrichtung 6.1 – Mathematik

Preprint Nr. 364

Matrix Valued Adaptive Cross Approximation

Sergej Rjasanow and Lucy Weggler

Saarbrücken 2015

Fachrichtung 6.1 – Mathematik
Universität des Saarlandes

Preprint No. 364
submitted: July 9, 2015

Matrix Valued Adaptive Cross Approximation

Sergej Rjasanow

Saarland University
Department of Mathematics
P.O. Box 15 11 50
66041 Saarbrücken
Germany
rjasanow@num.uni-sb.de

Lucy Weggler

BIOTRONIK SE & Co. KG
Woermannkehre 1
12359 Berlin
Germany
lucy.weggler@biotronik.com

Edited by
FR 6.1 – Mathematik
Universität des Saarlandes
Postfach 15 11 50
66041 Saarbrücken
Germany

Fax: + 49 681 302 4443
e-Mail: preprint@math.uni-sb.de
WWW: http://www.math.uni-sb.de/

Matrix Valued Adaptive Cross Approximation

S. Rjasanow and L. Weggler

July 9, 2015

Abstract

A new variant of the Adaptive Cross Approximation (ACA) for approximation of dense block matrices is presented. This algorithm can be applied to matrices arising from the Boundary Element Methods (BEM) for elliptic or Maxwell systems of partial differential equations. The usual interpolation property of the ACA is generalised for the matrix valued case. Some numerical examples demonstrate the efficiency of the new method. The main example will be the electromagnetic scattering problem, i.e. the exterior boundary value problem for the Maxwell system. Here, we will show that the matrix valued ACA method works well for high order BEM and the corresponding high rate of convergence is preserved. Another example shows the efficiency of the new method in comparison with the standard technique while approximating the smoothed version of the matrix valued fundamental solution of the time harmonic Maxwell system.

Keywords: Adaptive Cross Approximation, Maxwell system, Boundary Element Methods

1 Introduction

Dense matrices of high dimension appear in many different applications, however, one of the most important sources for such matrices are the Boundary Element Methods (BEM) for elliptic scalar equations, e.g. Laplace and Helmholtz equations, as well as for elliptic systems like the system of Lamé equations in elastostatics. However, one of the most challenging problems is the time harmonic system of Maxwell equations, i.e. the electromagnetic scattering problem.

Due to quadratic amount of memory, the classical boundary element realisations are applicable only for a rather moderate number N of boundary

elements. Fortunately, all boundary element matrices can be decomposed into a hierarchical system of blocks which can be approximated by the use of low rank matrices. This approximation can be computed by the use of the ACA-Algorithm. This is nowadays a well established numerical tool in both, scientific and commercial software. The first publications of the ACA are [2] in mathematical literature and [10] in an engineering journal. For the systems of partial differential equations, the fundamental solution is a matrix valued function leading to the block structure of the final system matrix. The number of blocks is there equal to the number of partial differential equations in the system under consideration and, therefore, is rather small. The dimension of the blocks, however, is big and proportional to the number of degrees of freedom. As a rule, its blocks will be approximated independently by the use of the conventional scalar valued ACA-Algorithm, see [3],[12] for the system of Lamé equations.
In this paper, we introduce a new version of the ACA which is designed to approximate block matrices having many rather small blocks with their block-rows and block-columns directly. The dimension of the small blocks is now equal to the number of partial differential equations and the number of the blocks is proportional to the number of boundary elements.
The article is organised as follows. A model problem of three-dimensional electromagnetic scattering and its numerical solution with a high order boundary element method is given in Section 2. In Section 3, we formulate a new ACA algorithm, state its properties and discuss the problem of the pivoting which is important for the practical realisation. Two numerical examples are presented in the final Section 4. The first example shows the numerical results obtained for a problem by approximating a smoothed version of the matrix valued fundamental solution. Here we show a comparison with the standard technique. The second example is an application of the method to a electromagnetic scattering problem, where the block structure results from the element matrices in a discontinuous Galerkin approach.

2 Model problem

A typical example, we have in mind, is a numerical solution of the three-dimensional electromagnetic scattering problem on a perfectly conducting body $\Omega \subset \mathbb{R}^3$. The scattering is initiated by an incoming plane wave of the form

$$\boldsymbol{E}^{\mathrm{inc}}(\boldsymbol{x}, t) = \boldsymbol{A} e^{\mathrm{i}(\boldsymbol{\kappa}\cdot\boldsymbol{x} - \omega t)} ,$$

where the angular frequency ω and the plane of propagation $\{\boldsymbol{\kappa}, \boldsymbol{A}\}$ are given. The physical quantity of interest is the scattered electric field

$$\boldsymbol{E} \in \boldsymbol{H}(\mathbf{curl}, \Omega^c)\,,$$

where $\Omega^c = \mathbb{R}^3 \setminus \overline{\Omega}$ is the exterior domain.

System of partial differential equations

The scattered electric field $\boldsymbol{E}$ is a weak solution of the exterior boundary value problem [11]

$$\left\{ \begin{array}{rcll} \mathbf{curl}\,\mathbf{curl}\,\boldsymbol{E} - \kappa^2 \boldsymbol{E} & = & \mathbf{0}\,, & \text{in } \Omega^c\,, \\ \gamma_D \boldsymbol{E} & = & \boldsymbol{m}\,, & \text{on } \Gamma = \partial\Omega\,, \end{array} \right. \tag{1}$$

where

$$\boldsymbol{m} = -\gamma_D \boldsymbol{E}^{\text{inc}} \in H^{-\frac{1}{2}}(\text{div}_\Gamma, \Gamma)$$

denotes the Dirichlet trace of the incoming wave on the boundary Γ. We assume that Ω^c is free-space and, therefore, its material parameters are

$$\varepsilon_0 = 8.854 \cdot 10^{-12}\ \frac{\text{As}}{\text{Vm}}\,, \quad \mu_0 = 4\pi \cdot 10^{-7}\ \frac{\text{Vs}}{\text{Am}}\,.$$

The wave number κ in (1) is then

$$\kappa = \omega\sqrt{\varepsilon_0 \mu_0}\,.$$

In addition, the solution $\boldsymbol{E}$ is subjected to the radiation condition at infinity

$$\left| \mathbf{curl}\,\boldsymbol{E}(\boldsymbol{x}) \times \frac{\boldsymbol{x}}{|\boldsymbol{x}|} - \mathrm{i}\,\omega\,\varepsilon_0\,\boldsymbol{E}(\boldsymbol{x}) \right| = \mathcal{O}\left(\frac{1}{|\boldsymbol{x}|^2}\right) \quad \text{for } |\boldsymbol{x}| \to \infty\,.$$

Fundamental solution

For $\kappa \neq 0$, the system of differential equations (1) admits the fundamental solution

$$K(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{4\pi} \frac{e^{-\mathrm{i}\kappa|\boldsymbol{x}-\boldsymbol{y}|}}{|\boldsymbol{x}-\boldsymbol{y}|} \cdot I + \frac{1}{\kappa^2} \mathcal{H}\left(\frac{1}{4\pi} \frac{e^{-\mathrm{i}\kappa|\boldsymbol{x}-\boldsymbol{y}|}}{|\boldsymbol{x}-\boldsymbol{y}|} \right) \in \mathbb{C}^{3\times 3}\,, \tag{2}$$

where $\mathcal{H}$ denotes the Hesse matrix of the scalar valued fundamental solution of the Helmholtz equation. Thus, for every $\boldsymbol{x} \neq \boldsymbol{y}$ the fundamental solution is a complex valued, symmetric 3 by 3 matrix. A smoothed version of the fundamental solution will be used in Section 4.1 for numerical tests.

Representation formula

Starting point for the numerical solution of the system (1) is the Stratton-Chu representation formula for $\boldsymbol{x} \in \Omega^c$

$$\boldsymbol{E}(\boldsymbol{x}) = \boldsymbol{S}_E(\boldsymbol{j})(\boldsymbol{x}) = \boldsymbol{S}_\kappa(\boldsymbol{j})(\boldsymbol{x}) + \frac{1}{\kappa^2}\nabla S_\kappa(\boldsymbol{j})(\boldsymbol{x})\,, \tag{3}$$

where

$$\boldsymbol{S}_\kappa(\boldsymbol{j})(\boldsymbol{x}) = \int_\Gamma \frac{e^{-\mathrm{i}\kappa|\boldsymbol{x}-\boldsymbol{y}|}}{4\pi|\boldsymbol{x}-\boldsymbol{y}|}\,\boldsymbol{j}(\boldsymbol{y})\,\mathrm{d}\sigma_{\boldsymbol{y}}$$

and

$$S_\kappa(\boldsymbol{j})(\boldsymbol{x}) = \int_\Gamma \frac{e^{-\mathrm{i}\kappa|\boldsymbol{x}-\boldsymbol{y}|}}{4\pi|\boldsymbol{x}-\boldsymbol{y}|}\,\mathrm{div}_\Gamma \boldsymbol{j}(\boldsymbol{y})\,\mathrm{d}\sigma_{\boldsymbol{y}}\,.$$

Thus, once the surface current $\boldsymbol{j}$ is found, the scattered electric field $\boldsymbol{E}$ can be determined everywhere by evaluating the surface potentials in (3).

Variational formulation

The surface current $\boldsymbol{j} \in H^{-\frac{1}{2}}(\mathrm{div}_\Gamma, \Gamma)$ is the unique solution (see [7, 4] for details) of the variational formulation

$$\langle \gamma_D \boldsymbol{S}_E(\boldsymbol{j}) \times \boldsymbol{n}, \boldsymbol{\phi}\rangle_{-\frac{1}{2}} = \langle \boldsymbol{m} \times \boldsymbol{n}, \boldsymbol{\phi}\rangle_{-\frac{1}{2}} \tag{4}$$

for all $\boldsymbol{\phi} \in H^{-\frac{1}{2}}(\mathrm{div}_\Gamma, \Gamma)$. In (4), $\boldsymbol{n}$ is the outward unit normal vector field on the boundary Γ and the brackets $\langle\cdot,\cdot\rangle$ denote the duality pairing

$$\langle\cdot,\cdot\rangle_{-\frac{1}{2}} \,:\, H^{-\frac{1}{2}}(\mathrm{curl}_\Gamma, \Gamma) \times H^{-\frac{1}{2}}(\mathrm{div}_\Gamma, \Gamma) \to \mathbb{C}\,.$$

The boundary integral equation that underlies (4) is referred to as the Electric Field Integral Equation (EFIE).

Galerkin BEM

The BEM determines an approximation $\boldsymbol{j}_{hp}$ of the current $\boldsymbol{j}$ in a finite-dimensional subspace

$$\boldsymbol{\mathcal{V}}_{hp}(\Gamma) \subset H^{-\frac{1}{2}}(\mathrm{div}_\Gamma, \Gamma)\,, \quad \dim \boldsymbol{\mathcal{V}}_{hp}(\Gamma) = N_{\mathrm{glob}}\,. \tag{5}$$

The Galerkin method is used to obtain a discretisation of the integral equation (4). Find $\boldsymbol{j}_{hp} \in \boldsymbol{\mathcal{V}}_{hp}(\Gamma)$ such that (4) is fulfilled for all $\boldsymbol{\phi} \in \boldsymbol{\mathcal{V}}_{hp}(\Gamma)$. Let

$$\{\boldsymbol{\phi}_\ell\}_{\ell=1}^{N_{\mathrm{glob}}} \tag{6}$$

be a basis in $\boldsymbol{\mathcal{V}}_{hp}(\Gamma)$ and let

$$\left(j_1, \ldots, j_{N_{\mathrm{glob}}}\right)^\top$$

denote the coefficients in the Galerkin ansatz for $\boldsymbol{j}_{hp}$. Then, the Galerkin discretisation of the integral equation simplifies to a set of N_{glob} linear equations

$$\sum_{\ell=1}^{N_{\mathrm{glob}}} \left(\gamma_D \boldsymbol{S}_E(\boldsymbol{\phi}_\ell) \times \boldsymbol{n}, \boldsymbol{\phi}_k\right)_\Gamma j_\ell = \left(\boldsymbol{m} \times \boldsymbol{n}, \boldsymbol{\phi}_k\right)_\Gamma \,, \quad k = 1, \ldots, N_{\mathrm{glob}} \,.$$

This system of linear equations can be equivalently written with a matrix-vector notation with Galerkin matrix

$$A \in \mathbb{C}^{N_{\mathrm{glob}} \times N_{\mathrm{glob}}}, \quad (A)_{k\ell} = \left(\gamma_D \boldsymbol{S}_E(\boldsymbol{\phi}_\ell), \boldsymbol{\phi}_k\right)_\Gamma \in \mathbb{C} \,. \tag{7}$$

A possible construction of the Galerkin basis (6) builds up on a set of Nédélec elements of the first kind. We shortly recall the construction procedure for the basis functions of an uniform order p and refer to [8] for a more detailed description. Let the surface Γ be given by a regular mesh consisting of curvilinear triangular boundary elements

$$\Gamma = \bigcup_{i=1}^{N} \overline{\Gamma_i} \,, \quad \boldsymbol{X}_i \colon \hat{T} \to \Gamma_i, \quad \boldsymbol{X}_i \in \mathcal{P}^p(\hat{T})^3 \,. \tag{8}$$

Thus, by assumption, the domain of definition of all parameterisations $\boldsymbol{X}_i$ of elements Γ_i is the reference triangle

$$\hat{T} = \left\{ \boldsymbol{\xi} = (\xi_1, \xi_2)^\top \in \mathbb{R}^2 \,,\ 0 < \xi_1, \xi_2 < 1,\ \xi_1 + \xi_2 < 1 \right\} \,. \tag{9}$$

The space $\mathcal{P}^p(\hat{T})^3$ contains all vector valued polynomials of degree p. On the reference triangle $\hat{T}$, a set of Nédélec elements of the first kind [14, 8] will be defined, i.e.

$$\boldsymbol{\mathcal{P}}_{1,p}(\hat{T}) \oplus \boldsymbol{\mathcal{P}}_{2,p}(\hat{T}) = \operatorname{span} \left\{ \hat{\boldsymbol{\varphi}}_k = (\hat{\varphi}_{k,1}, \hat{\varphi}_{k,2})^\top : \hat{T} \to \mathbb{R}^2 \right\}_{k=1}^{N_{\mathrm{loc}}} \,,$$

where

$$\boldsymbol{\mathcal{P}}_{1,p}(\hat{T}) = \left\{ \hat{\boldsymbol{\varphi}} \in \boldsymbol{\mathcal{P}}^{p-1}(\hat{T}),\ \hat{\boldsymbol{\varphi}}\big|_{\hat{e}_i} \cdot \hat{\boldsymbol{\tau}}_i \in \mathcal{P}^{p-1}(\hat{e}_i),\ i = 1, 2, 3 \right\}, \quad p \geq 1 \,,$$

$$\boldsymbol{\mathcal{P}}_{2,p}(\hat{T}) = \left\{ \hat{\boldsymbol{\varphi}} \in \boldsymbol{\mathcal{P}}^{p}(\hat{T}),\ \hat{\boldsymbol{\varphi}}\big|_{\hat{e}_i} \cdot \hat{\boldsymbol{\tau}}_i \in \mathcal{P}^{p-1}(\hat{e}_i),\ i = 1, 2, 3 \right\} , \quad p \geq 1 \,.$$

Here, we use the notation $\hat{e}_i$ for the edges of $\hat{T}$ and $\hat{\boldsymbol{\tau}}_i$ for the corresponding unit tangent vectors. A set of N_{loc} corresponding functions on the elements Γ_i is defined via Piola transformation, see [5]

$$\boldsymbol{\varphi}_k^i(\boldsymbol{x}) = T_i\, \hat{\boldsymbol{\varphi}}_k\left(\boldsymbol{X}_i^{-1}(\boldsymbol{x})\right), \quad T_i = \frac{1}{J_i}(-\boldsymbol{a}_{i,2} \vdots \boldsymbol{a}_{i,1}) \in \mathbb{R}^{3\times 2}, \quad k = 1, \dots, N_{\mathrm{loc}}, \tag{10}$$

where J_i denotes the Jacobian of the parameterisation $\boldsymbol{X}_i$ and $\boldsymbol{a}_{i,1}$, $\boldsymbol{a}_{i,2}$ are the natural tangent vector fields on Γ_i, i.e.,

$$\boldsymbol{a}_{i,l}(\boldsymbol{x}) = \frac{\partial(\boldsymbol{X}_i)(\boldsymbol{\xi})}{\partial \xi_l} \in \mathbb{R}^3, \quad l = 1, 2.$$

Thus, by Piola transformation, on each element Γ_i, we are given a set of N_{loc} the so-called local functions. A local function is non-polynomial and tangential on Γ and its support comprises only one element. The local functions allow for the definition of the Galerkin basis (6) spanning the finite-dimensional space $\boldsymbol{\mathcal{V}}_{hp}(\Gamma)$. Galerkin basis functions are partially continuous across the edges in the mesh Γ meaning that the scalar-valued function resulting from a projection with respect to the edge normal vector fields is continuous [6]. As usual, each local function contributes to a unique Galerkin basis function and, correspondingly, the restriction of the Galerkin basis function on a specific element in the mesh does either vanish or it coincides with a local function [8],[15]. Thus, there is an isomorphism ι relating the numbering of the local functions and the numbering of the globally defined Galerkin basis functions, namely,

$$\iota\colon (i, k_{\mathrm{loc}}) \mapsto k_{\mathrm{glob}} \quad \text{such that} \quad \boldsymbol{\varphi}^i_{k_{\mathrm{loc}}} = \boldsymbol{\phi}_{k_{\mathrm{glob}}}\big|_{\Gamma_i} \text{ with } k_{\mathrm{glob}} = \iota(i, k_{\mathrm{loc}}). \tag{11}$$

With help of the isomorphism ι the so-called connectivity matrix reads

$$P \in \mathbb{N}_0^{NN_{\mathrm{loc}}\times N_{\mathrm{glob}}}, \quad (P)_{i\,k_{\mathrm{loc}},k_{\mathrm{glob}}} = \begin{cases} \pm 1, & k_{\mathrm{glob}} = \iota(i, k_{\mathrm{loc}}), \\ 0, & k_{\mathrm{glob}} \neq \iota(i, k_{\mathrm{loc}}). \end{cases} \tag{12}$$

The sign in (12) is defined through the orientation of the edge in the element. Let us assume that the local functions are written in a vector and sorted element-wise, i.e.,

$$\left(\boldsymbol{\varphi}_1^1, \dots, \boldsymbol{\varphi}_{N_{\mathrm{loc}}}^1, \boldsymbol{\varphi}_1^2, \dots, \boldsymbol{\varphi}_{N_{\mathrm{loc}}}^2, \dots, \boldsymbol{\varphi}_1^N, \dots, \boldsymbol{\varphi}_{N_{\mathrm{loc}}}^N\right)^\top,$$

and define

$$D = \begin{pmatrix} D_{11} & D_{12} & \dots & D_{1N} \\ D_{21} & D_{22} & \dots & D_{2N} \\ \dots & \dots & \dots & \dots \\ D_{N1} & D_{N2} & \dots & D_{NN} \end{pmatrix} \in \mathbb{C}^{NN_{\mathrm{loc}}\times NN_{\mathrm{loc}}}. \tag{13}$$

where the so-called element matrices

$$D_{ij} \in \mathbb{R}^{N_{\text{loc}} \times N_{\text{loc}}}, \quad i, j = 1, \ldots, N \tag{14}$$

describe the interaction of the boundary elements Γ_i and Γ_j via their set of local functions. In the context of boundary element methods, the matrix D appears when a discontinuous Galerkin discretisation is considered because local functions are not related to each other and thus, they are globally discontinuous. However, by the connectivity matrix P the discontinuous Galerkin matrix D can be easily transformed into the Galerkin matrix A defined in (7). Namely, it holds

$$A = P^\top D P \,. \tag{15}$$

3 Matrix valued ACA algorithm

Since the fundamental solutions usually exhibit a singularity for $\boldsymbol{x} \to \boldsymbol{y}$, the standard procedure of hierarchical clustering of the initial degrees of freedom has to be applied at the beginning leading to a hierarchical matrix, [9] with admissible blocks. In what follows, we consider only an approximation of Trefftz like matrices. The corresponding generalisation for collocation or Galerkin matrices is obvious. Let X and Y be two non-empty, well separated subsets of the Euclidean space $\mathbb{R}^d$, where $d = 2, 3$ is the dimension of the physical space. Furthermore, let

$$K \,:\, X \times Y \to \mathbb{C}^{n \times n} \tag{16}$$

be a given matrix valued function of two $d-$dimensional variables $\boldsymbol{x} \in X$ and $\boldsymbol{y} \in Y$.

Algorithm 1.

1. `Initialisation`

$$R_0(\boldsymbol{x}, \boldsymbol{y}) = K(\boldsymbol{x}, \boldsymbol{y}), \quad S_0(\boldsymbol{x}, \boldsymbol{y}) = 0$$

2. `For` $k = 0, 1, \ldots$ `compute`

   2.1 `new pivot position`

$$(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = \text{PivotPosition}(R_k)$$

`2.2 new cross`

$$C_{k+1}(\boldsymbol{x},\boldsymbol{y}) = R_k(\boldsymbol{x},\boldsymbol{y}_{k+1})\left(R_k(\boldsymbol{x}_{k+1},\boldsymbol{y}_{k+1})\right)^{-1} R_k(\boldsymbol{x}_{k+1},\boldsymbol{y})$$

`2.3 new residuum`

$$R_{k+1}(\boldsymbol{x},\boldsymbol{y}) = R_k(\boldsymbol{x},\boldsymbol{y}) - C_{k+1}(\boldsymbol{x},\boldsymbol{y})$$

`2.4 new approximation`

$$S_{k+1}(\boldsymbol{x},\boldsymbol{y}) = S_k(\boldsymbol{x},\boldsymbol{y}) + C_{k+1}(\boldsymbol{x},\boldsymbol{y})$$

After $m \geq 1$ steps of the ACA-Algorithm 1, we obtain a sequence of residua $R_0, \ldots, R_m$ and a sequence of approximations $S_0, \ldots, S_m$ with the following properties.

1. Approximation property for $k = 0, \ldots, m$

$$R_k(\boldsymbol{x},\boldsymbol{y}) + S_k(\boldsymbol{x},\boldsymbol{y}) = K(\boldsymbol{x},\boldsymbol{y})\,,\ \boldsymbol{x} \in X\,, \boldsymbol{y} \in Y \tag{17}$$

2. Uniform interpolation property for $k = 1, \ldots, m$ and $\ell = 1, \ldots, k$

$$R_k(\boldsymbol{x},\boldsymbol{y}_\ell) = R_k(\boldsymbol{x}_\ell,\boldsymbol{y}) = 0 \quad \text{for all } \boldsymbol{x} \in X\,,\ \boldsymbol{y} \in Y \tag{18}$$

or

$$S_k(\boldsymbol{x},\boldsymbol{y}_\ell) = K(\boldsymbol{x},\boldsymbol{y}_\ell)\,,\ \boldsymbol{x} \in X\,, \quad S_k(\boldsymbol{x}_\ell,\boldsymbol{y}) = K(\boldsymbol{x}_\ell,\boldsymbol{y})\,,\ \boldsymbol{y} \in Y \tag{19}$$

3. Harmonicity property for $k = 0, \ldots, n$
   Let $\mathcal{L}_{\boldsymbol{x}}$ be a partial differential operator. If

$$\mathcal{L}_{\boldsymbol{x}} K(\boldsymbol{x},\boldsymbol{y}) = 0\,,\ \boldsymbol{x} \in \Omega$$

   then

$$\mathcal{L}_{\boldsymbol{x}} R_k(\boldsymbol{x},\boldsymbol{y}) = \mathcal{L}_{\boldsymbol{x}} S_k(\boldsymbol{x},\boldsymbol{y}) = 0\,,\ x \in \Omega$$

4. Non-recursive representation for $k = 1, \ldots, n$

$$S_k(\boldsymbol{x},\boldsymbol{y}) = u_k^\top(\boldsymbol{x}) V_k^{-1} w_k(\boldsymbol{y})\,,\ V_k \in \mathbb{R}^{n\,k \times n\,k}\,,\ u_k(\boldsymbol{x}), w_k(\boldsymbol{y}) \in \mathbb{R}^{n\,k \times k} \tag{20}$$

   with

$$u_k(\boldsymbol{x}) = \left(K(\boldsymbol{x},\boldsymbol{y}_0)^\top, \ldots, K(\boldsymbol{x},\boldsymbol{y}_{k-1})^\top\right)^\top,$$
$$w_k(\boldsymbol{y}) = \left(K(\boldsymbol{x}_0,\boldsymbol{y})^\top, \ldots, K(\boldsymbol{x}_{k-1},\boldsymbol{y})^\top\right)^\top$$

   and

$$V_k = \Big(K(\boldsymbol{x}_i,\boldsymbol{y}_j)\Big)_{i,j=1}^{k}$$

All of the above properties, except the last one, can be easily seen by induction. The first proof of the non-recursive representation is more technical and can be found in [2].
The most crucial step of Algorithm 1 is `2.1` where the position of the new pivot element is to be fixed. First of all, we have to guarantee the regularity of the matrix $R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})$. In general, it will be no problem. However, if R_k is regular at many positions, it is not clear how to choose the best one. In the scalar case, as a rule, we choose the position where the residuum R_k takes its maximal value (in modulus). In the matrix valued case, however, there are many possibilities. Let

$$R_k(\boldsymbol{x}, \boldsymbol{y}) = U_k(\boldsymbol{x}, \boldsymbol{y}) \Sigma_k(\boldsymbol{x}, \boldsymbol{y}) V_k^*(\boldsymbol{x}, \boldsymbol{y})$$

be the Singular Value Decomposition (SVD) of the residuum R_k with

$$\Sigma_k(\boldsymbol{x}, \boldsymbol{y}) = \operatorname{diag}\Big(\sigma_1^{(k)}(\boldsymbol{x}, \boldsymbol{y}), \ldots, \sigma_n^{(k)}(\boldsymbol{x}, \boldsymbol{y})\Big)$$

and

$$\sigma_1^{(k)}(\boldsymbol{x}, \boldsymbol{y}) \geq \sigma_2^{(k)}(\boldsymbol{x}, \boldsymbol{y}) \geq \ldots \geq \sigma_n^{(k)}(\boldsymbol{x}, \boldsymbol{y}) \geq 0\,.$$

The most obvious criterion is to maximise the largest singular value

$$(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = \operatorname{ArgMax} \sigma_1^{(k)}(\boldsymbol{x}, \boldsymbol{y})\,.$$

However, this criterion does not guarantee the regularity of the pivot matrix $R_k(\boldsymbol{x}, \boldsymbol{y})$ and, therefore, can not be used. The next possibility, which is attractive because of geometrical considerations, is to choose the matrix of "maximal volume" as the pivot element, i.e. to maximise the determinant

$$(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = \operatorname{ArgMax} \Big|\det R_k(\boldsymbol{x}, \boldsymbol{y})\Big| = \prod_{i=1}^{n} \sigma_i^{(k)}(\boldsymbol{x}, \boldsymbol{y})\,.$$

However, the residua R_k converge quite fast to zero and the corresponding criterion value will be close to the very small value $(\varepsilon_{\mathrm{ACA}})^n$ which can not be used in the practice even for the less critical case with $n = 3$ and $\varepsilon_{\mathrm{ACA}} = 10^{-6}$. The next possibility is based on the condition number of the matrix R_k

$$(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = \operatorname{ArgMin} \kappa_2\big(R_k(\boldsymbol{x}, \boldsymbol{y})\big) = \frac{\sigma_1^{(k)}(\boldsymbol{x}, \boldsymbol{y})}{\sigma_n^{(k)}(\boldsymbol{x}, \boldsymbol{y})}\,.$$

However, this criterion is relative, i.e. the criterion value will not converge to zero during the iterations. Thus, an important indicator for the ACA convergence will be lost. Furthermore, it is not clear why the identity matrix

diag(1, 1) is better pivot element than diag(2, 1). In this paper, we decide to use the following criterion for the choice of the pivot position

$$(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = \operatorname{ArgMax} \sigma_n^{(k)}(\boldsymbol{x}, \boldsymbol{y})\,.$$

In the pathological situation

$$\sigma_n^{(k)}(\boldsymbol{x}, \boldsymbol{y}) = 0 \quad \text{for all } \boldsymbol{x} \in X\,,\ \boldsymbol{y} \in Y\,,$$

there is no possibility to choose a regular pivot matrix $R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})$. Thus, the step `2.2` of Algorithm 1 needs a modification. One possibility is to use the Moore–Penrose pseudoinverse matrix $R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})$ instead of the non-existing inverse. However, the price for this modification is the loss of the uniform interpolation property (18) in this step, i.e.

$$\begin{aligned} R_{k+1}(\boldsymbol{x}_{k+1}, \boldsymbol{y}) &= R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}) - R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\, R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}) \\ &= \Big(I - R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\, R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\Big) R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}) \neq 0 \end{aligned}$$

and

$$\begin{aligned} R_{k+1}(\boldsymbol{x}, \boldsymbol{y}_{k+1}) &= R_k(\boldsymbol{x}, \boldsymbol{y}_{k+1}) - R_k(\boldsymbol{x}, \boldsymbol{y}_{k+1})\, R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) \\ &= R_k(\boldsymbol{x}, \boldsymbol{y}_{k+1}) \Big(I - R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\, R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\Big) \neq 0\,. \end{aligned}$$

Note, that the matrices

$$I - R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\, R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})$$

and

$$I - R_k^+(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\, R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})$$

are the orthogonal projectors onto the non-trivial subspaces

$$\ker R_k^*(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) \quad \text{and} \quad \ker R_k(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1})\,,$$

respectively. Nevertheless, one of the main properties of the Moore–Penrose pseudoinverse matrix A^+ of a matrix A, namely,

$$A\, A^+ A = A\,,$$

guarantee the interpolation property at the pivot position

$$R_{k+1}(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}) = 0\,,$$

and, what is even more important, the uniform interpolation property at the previous points is still valid, i.e. for $\ell = 1, \ldots, k$

$$R_{k+1}(\boldsymbol{x}_\ell, \boldsymbol{y}) = R_{k+1}(\boldsymbol{x}, \boldsymbol{y}_\ell) = 0\,, \quad \text{for all } \boldsymbol{x} \in X\,,\ \boldsymbol{y} \in Y\,.$$

A direct way to compute the Moore–Penrose pseudoinverse matrix A^+ of a singular matrix A is to use its SVD

$$A = U\,\Sigma\,V^*\,, \quad \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k > 0\,,\ \sigma_{k+1} = \ldots = \sigma_n = 0\,.$$

Thus, we get

$$A^+ = V\,\Sigma^+\,U^*\,,$$

where

$$\Sigma^+ = \operatorname{diag}\Big(\sigma_1^{-1}, \ldots, \sigma_k^{-1}, 0, \ldots, 0\Big)\,.$$

A further remark is that taking the pseudoinverse is not a continuous operation. An arbitrarily small change of a singular value from a zero value to non-zero leads to an arbitrarily big change in the pseudoinverse matrix. Thus, numerically it is necessary to work with some tolerance and to declare all singular values below this tolerance to be zeros.
Note, that in the regular situation, we have to solve a linear system with the matrix R_k. To this end, we will always use its SVD decomposition, which is the most stable solution technique for linear systems, since it is already computed for the choice of the pivot element. Therefore, no LU decomposition of the matrix R_k or other technique is necessary.
Finally, we remark that for reducible matrices, having the exact zero blocks, the usual control of the "column sum" and "row sum" is applied to prevent the situation where only a part of the matrix will be approximated. We skip the details.

4 Numerical examples

4.1 Fundamental solution

For the numerical tests, we first consider a smoothed version of the fundamental solution (2)

$$K_\alpha(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{4\pi}\frac{e^{-\mathrm{i}\kappa\sqrt{|\boldsymbol{x}-\boldsymbol{y}|^2+\alpha^2}}}{\sqrt{|\boldsymbol{x}-\boldsymbol{y}|^2+\alpha^2}} \cdot I + \frac{1}{\kappa^2}\mathcal{H}\left(\frac{1}{4\pi}\frac{e^{-\mathrm{i}\kappa\sqrt{|\boldsymbol{x}-\boldsymbol{y}|^2+\alpha^2}}}{\sqrt{|\boldsymbol{x}-\boldsymbol{y}|^2+\alpha^2}}\right) \in \mathbb{C}^{3\times 3}$$

with $\alpha = 0.1$, $\kappa = 5.0$ and a series of the discretisations of the unit sphere by N plane triangles as it is shown in Figure 4.1. Let $\boldsymbol{z}_i$ be the midpoint

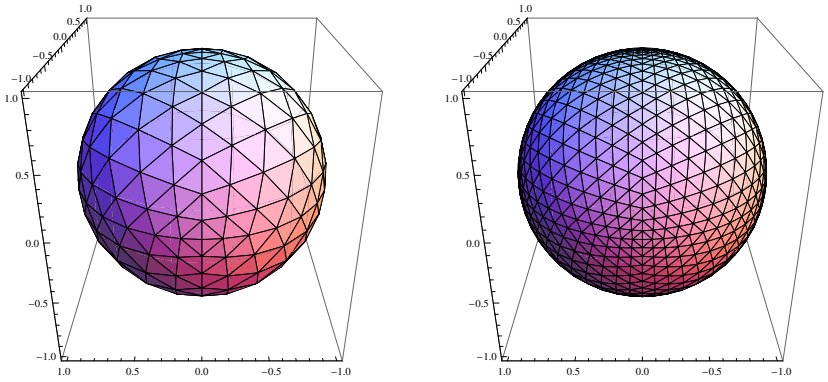

Figure 1: Discretisations of the unit sphere for $N = 512$ and $N = 2048$

Table 1: ACA approximation. Fundamental solution. Standard technique

N	M	$\varepsilon_{\mathrm{ACA}} = 10^{-2}$		$\varepsilon_{\mathrm{ACA}} = 10^{-4}$		$\varepsilon_{\mathrm{ACA}} = 10^{-6}$	
		MB	%	MB	%	MB	%
2048	1026	337.83	58.65%	345.46	59.98%	354.96	61.63%
8192	4098	1952.69	21.19%	2198.87	23.86%	2507.64	27.21%
32768	16386	8210.48	5.57%	10186.20	6.91%	12756.61	8.65%

of the boundary element Γ_i, $i = 1, \ldots, N$. The first possibility is to number the elements in such a way that the resulting matrix is of the form

$$D = \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{pmatrix},$$

where the big blocks

$$D_{ij} \in \mathbb{C}^{N \times N}, \quad i, j = 1, 2, 3$$

Table 2: ACA approximation. Fundamental solution. Matrix valued ACA

N	M	$\varepsilon_{\mathrm{ACA}} = 10^{-2}$		$\varepsilon_{\mathrm{ACA}} = 10^{-4}$		$\varepsilon_{\mathrm{ACA}} = 10^{-6}$	
		MB	%	MB	%	MB	%
2048	1026	278.32	48.32%	280.34	48.67%	282.82	49.10%
8192	4098	1405.44	15.25%	1569.48	17.03%	1789.75	19.42%
32768	16386	6030.95	4.09%	7299.07	4.95%	9039.05	6.13%

have the entries

$$\Big(D_{ij}\Big)_{k\ell} = \Big(K_\alpha(\boldsymbol{z}_k, \boldsymbol{z}_\ell)\Big)_{ij}\,, \quad i,j = 1,2,3\,,\ k,\ell = 1,\ldots,N\,.$$

The matrix is complex-symmetric. This property can be used, and only the six blocks $D_{11}, D_{12}, D_{13}, D_{22}, D_{23}$ and D_{33} will be approximated by the ACA algorithm. So far, this is the standard procedure for systems of partial differential equations, see [3],[12] for the system of Lamé equations and [13] for the system of Maxwell equations with high order boundary elements.
The second possibility is to number the DOF's in such a way that the matrix exhibits a block structure

$$D = \begin{pmatrix} D_{11} & D_{12} & \ldots & D_{1N} \\ D_{21} & D_{22} & \ldots & D_{2N} \\ \ldots & \ldots & \ldots & \ldots \\ D_{N1} & D_{N2} & \ldots & D_{NN} \end{pmatrix}\,,$$

where the small matrices

$$D_{ij} \in \mathbb{C}^{3\times 3}\,, \quad i,j = 1,\ldots,N$$

have the entries

$$\Big(D_{ij}\Big)_{k\ell} = \Big(K_\alpha(\boldsymbol{z}_i, \boldsymbol{z}_j)\Big)_{k\ell}\,, \quad i,j = 1,\ldots,N\,,\ k,\ell = 1,2,3\,.$$

Then the matrix valued ACA algorithm presented in Section 3 can be applied. The results of our tests are presented in Tables 1 and 2. It is clearly seen that the matrix valued ACA-Algorithm works better for all dimensions of the matrix. An explanation for this fact can be the size of the matrix. It is well known that the ACA suffers from the dimension. The standard technique, however, divides the whole big matrix in a set of smaller matrices which are approximated independently.

4.2 Galerkin BEM

In this section, the matrix valued ACA algorithm from Section 3 is applied to solve electromagnetic scattering problems. The electromagnetic scattering has been considered in Section 2 and a high order BEM has been introduced. Moreover, special attention has been paid to explain that the conventional Galerkin matrix A allows for a reformulation in terms of the discontinuous Galerkin matrix D, (13). A closer look at this matrix makes clear that D exhibits a structure which allows the application of the matrix valued ACA

Table 3: Galerkin BEM with $p = 1$. Matrix valued ACA.

N	N_{loc}	N_{glob}	ε_{ACA}	MB	%	$\varepsilon_{\text{GMRES}}$	Iter	Error
2048	3	3072	10^{-4}	280.00	48.55%	10^{-6}	254	$1.51 \cdot 10^{-1}$
8192	3	12288	10^{-5}	1563.00	16.96%	10^{-8}	509	$6.38 \cdot 10^{-2}$
32768	3	49152	10^{-6}	7631.00	5.18%	10^{-10}	1008	$3.13 \cdot 10^{-2}$

Table 4: Galerkin BEM with $p = 2$. Matrix valued ACA.

N	N_{loc}	N_{glob}	ε_{ACA}	MB	%	$\varepsilon_{\text{GMRES}}$	Iter	Error
2048	8	10240	10^{-4}	1848.00	45.12%	10^{-6}	611	$1.08 \cdot 10^{-2}$
8192	8	40960	10^{-5}	9584.00	14.62%	10^{-8}	1262	$2.77 \cdot 10^{-3}$
32768	8	163840	10^{-6}	44028.00	4.20%	10^{-10}	2295	$7.00 \cdot 10^{-4}$

driven by the small element matrices (14). Thus, the idea is to approximate the discontinuous Galerkin matrix D by matrix valued ACA and enforce the connectivity of the local functions within the iterative solver by (15). We consider a plane wave with complex amplitude

$$\boldsymbol{E}^{\text{inc}}(\boldsymbol{x}) = e^{-\mathrm{i}\kappa x_3}\boldsymbol{e}_1, \quad \kappa = 5.0\,.$$

This plane wave impinges on a perfectly conducting scatterer of spherical shape, see Figure 1. The problem has the analytic solution and explicit formulae for the surface current $\boldsymbol{j}$ are found in [1], for instance. Thus the error of the numerical solution can be carefully computed.
Numerical results for several Galerkin BEM simulations with varying order of approximations, $p = 1, 2, 3$, are shown in Tables 3–5. Note that the compression result refers to the total dimension of the discontinuous Galerkin matrix which is NN_{loc}. The complex symmetry of the matrix D is also exploited for the approximation. In order to keep the convergence of the high order BEM, we increase the accuracy of the ACA approximation with the dimension of the matrix. Thus, the linear, quadratic and even cubic convergence can be clearly seen in the last columns of Tables 3–5.
Finally, we remark that all computations with small $N_{\text{loc}} \times N_{\text{loc}}$ matrices are

Table 5: Galerkin BEM with $p = 3$. Matrix valued ACA.

N	N_{loc}	N_{glob}	ε_{ACA}	MB	%	$\varepsilon_{\text{GMRES}}$	Iter	Error
2048	24	21504	10^{-6}	6187.00	42.97%	10^{-8}	3767	$5.20 \cdot 10^{-4}$
8192	24	86016	10^{-8}	35300.00	15.32%	10^{-10}	7956	$6.58 \cdot 10^{-5}$

done by the use of the very effective third level BLAS and LAPACK libraries. The computational times for all the SVD decompositions and for the solution of the small linear systems is not really measurable and the change from $N_{\mathrm{loc}} = 3$ in Table 2 to $N_{\mathrm{loc}} = 24$ in Table 5 does not exhibit a real cubic behaviour. This can be seen in LAPACK starting from the dimension about 10 000. The total computational time is by the matrix valued version of the ACA is, as by the scalar version too, more or less proportional to the memory. Thus, in the first example, the new version is about 20% faster than the standard technique.

5 Conclusions

A new version of the ACA algorithm for dense matrices which have an additional block structure is introduced and tested. The algorithm fits perfectly to the high order discontinuous Galerkin philosophy of discretisation. It works well for different orders p of the boundary elements and guarantees the corresponding high rate of convergence for the exterior electromagnetic scattering problem. Furthermore, in comparison with the standard technique for block matrices, i.e. approximation of big blocks independently, the new scheme appears to consume less memory.

References

[1] C. Balanis. *Advanced Engineering Electromagnetics, Engineering.* Wiley, New York, 1989.

[2] M. Bebendorf. Approximation of boundary element matrices. *Numer. Math.*, 86(4):565–589, 2000.

[3] M. Bebendorf and R. Grzhibovskis. Accelerating Galerkin BEM for Linear Elasticity using Adaptive Cross Approximation. *Math. Meth. Appl. Sci.*, 29:1721–1747, 2006.

[4] A. Bespalov, N. Heuer, and R. Hiptmair. Convergence of the natural *hp*-BEM for the electric field integral equation on polyhedral surfaces. *SIAM J. Numer. Anal.*, 48(4):1518–1529, 2010.

[5] F. Brezzi and M Fortin. *Mixed and Hybrid Finite Element Methods.* Springer, New York, 1991.

[6] A. Buffa and P. Ciarlet,Jr. On traces for functional spaces related to Maxwell's equations. I. An integration by parts formula in Lipschitz polyhedra. *Math. Methods Appl. Sci.*, 24(1):9–30, 2001.

[7] A. Buffa, R. Hiptmair, T. von Petersdorff, and C. Schwab. Boundary element methods for Maxwell transmission problems in Lipschitz domains. *Numer. Math.*, 95(3):459–485, 2003.

[8] L. Demkowicz. *Computing with hp-adaptive finite elements. Vol. 1.* Applied Mathematics and Nonlinear Science Series. Chapman & Hall/CRC, Boca Raton, FL, 2007.

[9] W. Hackbusch. A sparse matrix arithmetic based on $\mathcal{H}$-matrices. I. Introduction to $\mathcal{H}$-matrices. *Computing*, 62(2):89–108, 1999.

[10] S. Kurz, O. Rain, and S. Rjasanow. The Adaptive Cross Approximation Technique for the 3D Boundary Element Method. *IEEE Transaction on Magnetics*, 38(2):421–424, 2002.

[11] J. C. Nédélec. *Acoustic and Electromagnetic Equations.* Springer, New York, 2001.

[12] S. Rjasanow and O. Steinbach. *The Fast Solution of Boundary Integral Equations.* Number 12 in Springer Series in Mathematical and Analytical Technology with Applications to Engineering. Springer-Verlag, Berlin-Heidelberg-NewYork, 2007.

[13] S. Rjasanow and L. Weggler. ACA accelerated high order BEM for Maxwell problems. *Comput. Mech.*, 51(4):431–441, 2013.

[14] J. Schöberl and S. Zaglmayr. High order Nédélec elements with local complete sequence properties. *COMPEL*, 24(2):374–384, 2005.

[15] L. Weggler. *High order Boundary Eelement Methods*. PhD thesis, University of Saarland, 2011.