Fast Multipole Methods and Applications

Günther Of

Institut für Numerische Mathematik, Technische Universität Graz,
Steyrergasse 30, 8010 Graz, Austria
of@tugraz.at

Summary. The symmetric formulation of boundary integral equations and the Galerkin boundary element method are considered to solve mixed boundary value problems of three–dimensional linear elastostatics. Fast boundary element techniques, like the fast multipole method, have to be used to overcome the quadratic complexity of standard boundary element methods. The fast methods provide a data sparse approximation of the fully populated matrices and reduce the computational costs and memory requirements from quadratic order to almost linear ones. Three different approaches to realize the boundary integral operators of linear elastostatics by the fast multipole method are described and numerical examples are given for one of these approaches.

1 Introduction

The Galerkin boundary element method for the symmetric formulation of boundary integral equations is an efficient and reliable tool to solve mixed boundary value problems in linear elastostatics by numerical simulations. This approach is based on a rigorous mathematical analysis. The related stability and error analysis can be found, for example, in [15, 16]. Mathematical books on boundary element methods are, for example, [17, 21, 35, 42].

As the involved boundary integral operators are non–local, standard boundary element techniques result in fully populated stiffness matrices. Therefore, standard boundary element methods are restricted to rather small problem sizes. Hence, fast boundary element methods have to be used for problems of engineering and industrial interest. There exist several fast boundary element methods reducing the memory requirements and the computational costs for a matrix times vector multiplication to almost linear complexity. Most of these methods rely on a clustering of the boundary elements. This leads to a block clustering of the considered matrix, too. Then low rank approximations are used for an appropriate approximation of the corresponding block matrices. The methods mainly differ in the construction and the realization of the low rank approximations. Among them there are the fast multipole

method [7, 8, 34] and the panel clustering method [12] which both perform the low rank approximation by an approximation of the kernel by appropriate series expansions providing a separation of the variables. The panel clustering method uses the Taylor series expansion whereas the fast multipole method uses spherical harmonics. The adaptive cross approximation (ACA) method [1, 33] is an algebraic approach to construct low rank approximations. The $\mathcal{H}$–matrices [10] provide a complete arithmetic for the class of matrices with low rank approximations. The $\mathcal{H}^2$–matrices [11] use hierarchical basis functions for the low rank approximations. The wavelet approximation methods [4] construct special nested trial spaces which enable a sparse approximation of the matrix due to the rapid decay of the kernel.

An extensive overview is given in [25] for the large number of contributions to the fast multipole method. There exist several versions of the realization of the boundary integral operators by the fast multipole method in three–dimensional linear elastostatics. A fast multipole version based on the reformulation of the kernel with respect to the fast multipole method for the Laplacian is given in [6]. There, the kernels of the boundary integral operators of linear elastostatics are decomposed in terms depending on $|x-y|^{-1}$ and its derivatives. Then the fast multipole method for electrostatics problems is used as a black box. This approach leads to a rather large number of applications of the potential theoretic fast multipole method.

Starting from the kernel expansion of the fundamental solution of the Laplacian, a new multipole expansion together with the corresponding translations and conversions are derived in [46] for the fundamental solution of linear elastostatics. That leads to less applications of the fast multipole method, but the expansions and the operations get more costly. The authors do not make a clear statement in [46] whether their approach is faster than the approach presented in [6]. The same expansion in spherical harmonics is presented for the panel clustering method in linear elastostatics in [14].

In [32], a different approach based on Taylor series expansions, which is easier to adopt to other kernels, is used. This version of the fast multipole method lacks the translations of local expansions from the clusters to their sons and converts the multipole expansions to the clusters of the finest level. Therefore, the number of conversions is rather high.

We have presented a fast multipole method for linear elastostatics in [29]. The realization of the single layer potential is similar to the approach presented in [6] but guarantees the symmetry of the approximation of the Galerkin matrix of the single layer potential. Integration by parts is used to reduce the hypersingular operator, the double layer potential and its adjoint to double layer potentials of the Laplacian and to single layer potentials.

Here, we try to give an review of some approaches to realize a fast boundary element methods for three–dimensional linear elastostatics based on the fast multipole method. First, we describe the symmetric formulation and the considered Galerkin discretization in Sect. 2. In Sect. 3, the fast multipole method is introduced independent of the specific kernel expansions which are

used in the approaches. These approaches are described and in parts compared in Sect. 4. Finally, several numerical examples are given in Sect. 5.

2 Symmetric Boundary Integral Formulation and Boundary Element Method

Let $\Omega \subset \mathbb{R}^3$ be a bounded, simply connected domain with a piecewise continuous Lipschitz boundary $\Gamma = \partial\Omega$, where the outer normal vector $n(x)$ is given for almost all $x \in \Gamma$. We consider a mixed boundary value problem of linear elastostatics, to determine the displacement field $u(x)$ for $x \in \Omega$,

$$\begin{aligned} -\operatorname{div} \sigma(u,x) &= 0 && \text{for } x \in \Omega, \\ \gamma_0 u(x) &= g_D(x) && \text{for } x \in \Gamma_D, \\ \gamma_1 u(x) &= g_N(x) && \text{for } x \in \Gamma_N. \end{aligned} \tag{1}$$

The boundary $\Gamma = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ is decomposed in disjoint parts Γ_D and Γ_N. This decomposition may be given componentwise. To guarantee the unique solvability of the boundary value problem, we assume that the part with Dirichlet boundary conditions must not vanish in each component, i.e., $\operatorname{meas}(\Gamma_{D,i}) > 0$. The stress tensor $\sigma(u)$ is related to the strain tensor $e(u)$ by Hooke's law

$$\sigma(u) = \frac{E\nu}{(1+\nu)(1-2\nu)} \operatorname{tr} e(u) I + \frac{E}{(1+\nu)} e(u)$$

with the Young modulus $E > 0$ and the Poisson ratio $\nu \in (-1, 1/2)$. For the case $\nu \to 1/2$, special techniques [40] have to be applied for the boundary element method. These techniques will not be addressed here. The strain tensor is defined by

$$e(u) = \frac{1}{2}(\nabla u^\top + \nabla u).$$

The trace operators are given by

$$\begin{aligned} \gamma_0 u(x) &:= \lim_{\Omega \ni \widetilde{x} \to x \in \Gamma} u(\widetilde{x}) && \text{for almost all } x \in \Gamma, \\ \gamma_1 u(x) &:= \lim_{\Omega \ni \widetilde{x} \to x \in \Gamma} [\sigma(u,\widetilde{x}) n(x)] && \text{for almost all } x \in \Gamma. \end{aligned}$$

The solution of the mixed boundary value problem (1) is given by the representation formula

$$u(x) = \int_\Gamma \gamma_{0,y} U^*(x,y) \gamma_1 u(y) ds_y - \int_\Gamma \left(\gamma_{1,y} U^*(x,y)\right)^\top \gamma_0 u(y) ds_y \tag{2}$$

for $x \in \Omega$. The fundamental solution of linear elastostatics is given by the Kelvin tensor

$$U^*_{ij}(x,y) = \frac{1}{8\pi}\frac{1}{E}\frac{1+\nu}{1-\nu}\left[\frac{(3-4\nu)}{|x-y|}\delta_{ij} + \frac{(x_i-y_i)(x_j-y_j)}{|x-y|^3}\right] \tag{3}$$

for $i,j = 1,\ldots,3$. The application of the trace operators to the representation formula gives the boundary integral equation

$$\gamma_0 u(x) = \int_\Gamma U^*(x,y)\gamma_1 u(y)ds_y + \frac{1}{2}\gamma_0 u(x) - \int_{\Gamma\setminus\{x\}} T^*(x,y)\gamma_0 u(y)ds_y,$$

for almost all $x \in \Gamma$ with $T^*(x,y) = (\gamma_{1,y}U^*(x,y))^\top$, and the hypersingular boundary integral equation

$$\gamma_1 u(x) = \frac{1}{2}\gamma_1 u(x) + \int_{\Gamma\setminus\{x\}} \gamma_{1,x}U^*(x,y)\gamma_1 u(y)ds_y - \gamma_{1,x}\int_\Gamma T^*(x,y)\gamma_0 u(y)ds_y$$

for almost all $x \in \Gamma$, respectively. Both boundary integral equations together form a system of boundary integral equations

$$\begin{pmatrix}\gamma_0 u\\ \gamma_1 u\end{pmatrix} = \begin{pmatrix}\frac{1}{2}I - K & V\\ D & \frac{1}{2}I + K'\end{pmatrix}\begin{pmatrix}\gamma_0 u\\ \gamma_1 u\end{pmatrix}. \tag{4}$$

In this representation, we use the standard notations for the boundary integral operators, in particular the single layer potential

$$(Vt)(x) = \int_\Gamma U^*(x,y)t(y)ds_y \qquad \text{for } x \in \Gamma,$$

the double layer potential

$$(Ku)(x) = \int_{\Gamma\setminus\{x\}} T^*(x,y)u(y)ds_y \qquad \text{for } x \in \Gamma,$$

its adjoint operator

$$(K't)(x) = \int_{\Gamma\setminus\{x\}} \gamma_{1,x}U^*(x,y)t(y)ds_y \qquad \text{for } x \in \Gamma,$$

and the hypersingular operator

$$(Du)(x) = -\gamma_{1,x}\int_\Gamma T^*(x,y)u(y)ds_y \qquad \text{for } x \in \Gamma.$$

Here and in what follows, t denotes the traction.

As the solution of the boundary value problem (1) is given by the representation formula (2), the complete Cauchy data $\gamma_0 u$ and $\gamma_1 u$ are sufficient for the evaluation of the solution $u(x)$ for $x \in \Omega$. It remains to determine $\gamma_0 u$ on Γ_N and $\gamma_1 u$ on Γ_D. First, suitable expansions $\widehat{g}_D \in H^{1/2}(\Gamma)$ and $\widehat{g}_N \in H^{-1/2}(\Gamma)$ of the given boundary data $g_D \in H^{1/2}(\Gamma_D)$ and $g_N \in H^{-1/2}(\Gamma_N)$ to the whole boundary Γ are chosen such that

$$\widehat{g}_D(x) = g_D(x) \quad \text{for } x \in \Gamma_D \quad \text{and} \quad \widehat{g}_N(x) = g_N(x) \quad \text{for } x \in \Gamma_N$$

hold. With the splitting of the Cauchy data into the known and the unknown parts,

$$\gamma_0 u(x) = \widehat{u}(x) + \widehat{g}_D(x) \quad \text{and} \quad \gamma_1 u(x) = \widehat{t}(x) + \widehat{g}_N(x),$$

only the functions $\widehat{u} \in \widetilde{H}^{1/2}(\Gamma_N)$ and $\widehat{t} \in \widetilde{H}^{-1/2}(\Gamma_D)$ have to be determined. The Sobolev space $\widetilde{H}^{1/2}(\Gamma_N)$ is the subset of functions in $H^{1/2}(\Gamma)$ with support on Γ_N. $\widetilde{H}^{-1/2}(\Gamma_D)$ is defined by duality of $H^{1/2}(\Gamma_D)$. The complete system (4) of the two boundary integral equations is used to determine the unknown functions $\widehat{u}$ and $\widehat{t}$. The use of the first boundary integral equation for $x \in \Gamma_D$ and of the hypersingular boundary integral equation for $x \in \Gamma_N$ ends up in the symmetric formulation [3, 39]:

$$(V\widehat{t})(x) - (K\widehat{u})(x) = (\frac{1}{2}I + K)\widehat{g}_D(x) - (V\widehat{g}_N)(x) \qquad \text{for } x \in \Gamma_D,$$

$$(K'\widehat{t})(x) + (D\widehat{u})(x) = (\frac{1}{2}I - K')\widehat{g}_N(x) - (D\widehat{g}_D)(x) \qquad \text{for } x \in \Gamma_N.$$

The equivalent variational formulation is given by:
Find $(\widehat{u}, \widehat{t}) \in \widetilde{H}^{1/2}(\Gamma_N) \times \widetilde{H}^{-1/2}(\Gamma_D)$, such that

$$a(\widehat{u}, \widehat{t}; v, \tau) = f(v, \tau) \qquad \text{for all } (v, \tau) \in \widetilde{H}^{1/2}(\Gamma_N) \times \widetilde{H}^{-1/2}(\Gamma_D) \qquad (5)$$

holds. The bilinear form is given by

$$a(\widehat{u}, \widehat{t}; v, \tau) = \langle V\widehat{t}, \tau\rangle_{\Gamma_D} - \langle K\widehat{u}, \tau\rangle_{\Gamma_D} + \langle K'\widehat{t}, v\rangle_{\Gamma_N} + \langle D\widehat{u}, v\rangle_{\Gamma_N},$$

and the linear form is defined by

$$\begin{aligned} f(v, \tau) = {} & \langle (\frac{1}{2}I + K)\widehat{g}_D, \tau\rangle_{\Gamma_D} - \langle V\widehat{g}_N, \tau\rangle_{\Gamma_D} \\ & + \langle (\frac{1}{2}I - K')\widehat{g}_N(x), v\rangle_{\Gamma_N} - \langle D\widehat{g}_D, v\rangle_{\Gamma_N}. \end{aligned}$$

The boundedness and the ellipticity of the bilinear form $a(\cdot;\cdot)$ on $\widetilde{H}^{1/2}(\Gamma_N) \times \widetilde{H}^{-1/2}(\Gamma_D)$ can be proofed by the boundedness of the operators and the ellipticity of the single layer potential V and of the hypersingular operator D on $\widetilde{H}^{-1/2}(\Gamma_D)$ and $\widetilde{H}^{1/2}(\Gamma_N)$. The unique solvability of the variational formulation (5) then follows by the Lemma of Lax–Milgram for the continuous linear form $f(\cdot)$.

Let the boundary $\Gamma = \partial\Omega$ be described by a union $\bigcup_{\ell=1}^{N} \overline{\tau}_\ell$ of plane triangles τ_ℓ with a local meshsize

$$h_\ell := \left(\int_{\tau_\ell} ds_x \right)^{1/2}.$$

The global meshsize is defined by

$$h := \max_{\ell=1,\dots,N} h_\ell.$$

Here, we consider a shape regular and quasi uniform boundary discretization for simplicity. We further assume that each boundary element τ_ℓ belongs either to $\Gamma_{D,i}$ or $\Gamma_{N,i}$ for each component $i = 1, \dots, 3$. For each component $k = 1, \dots, 3$, we use the finite–dimensional trial spaces

$$S_h^1(\Gamma_{N,k}) = \operatorname{span}\{\varphi_{i,k}\}_{i=1}^{M_{N,k}} \subset \widetilde{H}^{1/2}(\Gamma_{N,k})$$
$$S_h^0(\Gamma_{D,k}) = \operatorname{span}\{\psi_{j,k}\}_{j=1}^{N_{D,k}} \subset \widetilde{H}^{-1/2}(\Gamma_{D,k})$$

for the Galerkin discretization of the variational formulation (5) of the symmetric formulation. $S_h^1(\Gamma_N)$ is the space of the piecewise linear and continuous functions with support in Γ_N and is used for the approximation of the displacements u. The basis functions φ_i are the linear functions that are one in the node x_i of the boundary element mesh and zero in all other nodes. $S_h^0(\Gamma_D)$ denotes the space of piecewise constant functions with support in Γ_D and is used for the approximation of the tractions t. The basis functions ψ_j are one on the boundary element τ_j and zero on all others. N is the number of boundary elements and M is the number of nodes. An index restricts these numbers to the corresponding part of the boundary in the denoted component. For the componentwise trial functions

$$\widehat{u}_{h,k}(x) = \sum_{i=1}^{M_{N,k}} \widehat{u}_{i,k}\varphi_{i,k}(x) \quad \text{and} \quad \widehat{t}_{h,k}(x) = \sum_{j=1}^{N_{D,k}} \widehat{t}_{j,k}\psi_{j,k}(x),$$

we have to find the solution $(\widehat{u}_h, \widehat{t}_h) \in S_h^1(\Gamma_N) \times S_h^0(\Gamma_D)$ of the discrete Galerkin variational formulation

$$a(\widehat{u}_h, \widehat{t}_h; v_h, \tau_h) = f(v_h, \tau_h) \qquad \text{for all } (v_h, \tau_h) \in S_h^1(\Gamma_N) \times S_h^0(\Gamma_D). \tag{6}$$

It can be shown by means of Cea's lemma and the approximation properties of the trial spaces that the discrete variational formulation is uniquely solvable and that the following error estimate holds for the approximations $\widehat{u}_h$ and $\widehat{t}_h$ of the solutions $\widehat{u} \in \widetilde{H}^{s+1}(\Gamma_N)$ and $\widehat{t} \in \widetilde{H}^s_{\mathrm{pw}}(\Gamma_D)$ of the variational formulation (5):

$$\|\widehat{u} - \widehat{u}_h\|^2_{H^{1/2}(\Gamma)} + \|\widehat{t} - \widehat{t}_h\|^2_{H^{-1/2}(\Gamma)} \le ch^{2s+1}\left[\|\widehat{u}\|^2_{H^{s+1}(\Gamma)} + \|\widehat{t}\,\|^2_{H^s_{\mathrm{pw}}(\Gamma)}\right] \tag{7}$$

for $-1/2 \le s \le 1$, in the case of suitable extensions $\widehat{g}_D \in H^{s+1}(\Gamma)$ and $\widehat{g}_N \in H^s_{\mathrm{pw}}(\Gamma)$ of the boundary data. Here, $H^s_{\mathrm{pw}}(\Gamma)$ denotes an appropriately defined piecewise Sobolev space, see [42]. In the case of a sufficiently smooth solution, i.e., $s = 1$, an optimal convergence rate of $3/2$ is obtained in the energy norm.

The equivalent system of linear equations of the discrete variational formulation (6) is given by

$$\begin{pmatrix} V_h & -K_h \\ K_h' & D_h \end{pmatrix} \begin{pmatrix} \widehat{\underline{t}} \\ \widehat{\underline{u}} \end{pmatrix} = \begin{pmatrix} \underline{f}^1 \\ \underline{f}^2 \end{pmatrix}. \tag{8}$$

$\widehat{\underline{t}} \in \mathbb{R}^{N_D}$ and $\widehat{\underline{u}} \in \mathbb{R}^{M_N}$ with $N_D = N_{D,1} + N_{D,2} + N_{D,3}$ and $M_N = M_{N,1} + M_{N,2} + M_{N,3}$ are the vectors of the coefficients $\widehat{t}_\ell$ and $\widehat{u}_j$ of the trial functions $\widehat{t}_h$ and $\widehat{u}_h$. The block matrices are given from the discretization of the corresponding boundary integral operators on the corresponding parts of the boundary for $i, j = 1, \ldots, 3$ by

$$\begin{aligned}
V_h^{ij}[\ell, k] &= \langle V\psi_{k,j}, \psi_{\ell,i}\rangle_{\Gamma_D} && \text{for } \ell = 1, \ldots, N_{D,i}, k = 1, \ldots, N_{D,j}, \\
K_h^{ij}[\ell, k] &= \langle K\varphi_{k,j}, \psi_{\ell,i}\rangle_{\Gamma_D} && \text{for } \ell = 1, \ldots, N_{D,i}, k = 1, \ldots, M_{N,j}, \\
{K'}_h^{ij}[\ell, k] &= \langle K'\psi_{k,j}, \varphi_{\ell,i}\rangle_{\Gamma_N} && \text{for } \ell = 1, \ldots, M_{N,i}, k = 1, \ldots, N_{D,j}, \\
D_h^{ij}[\ell, k] &= \langle D\varphi_{k,j}, \varphi_{\ell,i}\rangle_{\Gamma_N} && \text{for } \ell = 1, \ldots, M_{N,i}, k = 1, \ldots, M_{N,j}.
\end{aligned}$$

The vectors of the right hand side are given by

$$\begin{aligned}
f_{\ell,i}^1 &= \langle (\frac{1}{2}I + K)\widehat{g}_D - V\widehat{g}_N, \psi_{\ell,i}\rangle_{\Gamma_D} && \text{for } \ell = 1, \ldots, N_{D,i}, \\
f_{k,i}^2 &= \langle (\frac{1}{2}I - K')\widehat{g}_N - D\widehat{g}_D, \varphi_{j,i}\rangle_{\Gamma_N} && \text{for } k = 1, \ldots, M_{N,i}.
\end{aligned}$$

The matrix of the system of linear equations (8) is block skew symmetric and positive definite. Furthermore, all blocks are fully populated, i.e., the memory requirements and the effort for one matrix times vector multiplication is of order $\mathcal{O}(N_D^2 + M_N^2)$.

3 Fast Multipole Boundary Element Methods

In this section, we describe the realization of the matrix times vector multiplication $\underline{w} = A_h\underline{t}$ or componentwise,

$$w_\ell = \sum_{k=1}^{\widetilde{N}} A_h[\ell, k]t_k = \sum_{k=1}^{\widetilde{N}} \int_\Gamma (A\varphi_k)(x)\psi_\ell(x)t_k ds_x \quad \text{for all } \ell = 1, \ldots, \widetilde{M}, \tag{9}$$

of some boundary integral operator

$$(At)(x) = \int_\Gamma \mathcal{Q}_x \mathcal{Q}_y k(x, y)t(y)ds_y$$

by the fast multipole method. $\mathcal{Q}_x$ and $\mathcal{Q}_y$ are some operators like linear combinations of partial derivatives operating on a kernel $k(x, y)$ with respect to x

and y, respectively. But these operators can also be just identities. $\{\varphi_k\}_{k=1}^{\widetilde{N}}$ is the set of trial functions and $\{\psi_\ell\}_{\ell=1}^{\widetilde{M}}$ is the set of test functions. These two sets might coincide. The basis functions φ_k and ψ_ℓ do not have to coincide with the definitions of Sect. 2. The main ingredients of the fast multipole method are the approximation of the kernel function by an appropriate series expansion and the use of a hierarchical structure to compute these expansions efficiently. We require that the kernel $k(x, y)$ is separable, i.e., an expansion

$$k(x, y) = \sum_{n=0}^{\infty} g_n(y) f_n(x)$$

exists with a separation of the variables x and y. Such an expansion can be given by a Taylor series expansion, for example. An approximation of the kernel is defined by truncating the infinite sum at a suitable chosen expansion degree $\tilde{p}$,

$$k_{\tilde{p}}(x, y) = \sum_{n=0}^{\tilde{p}} g_n(y) f_n(x). \tag{10}$$

If such a splitting of the kernel and the approximation (10) were valid for all x and y, the approximation of the matrix times vector multiplication (9) could be rewritten as

$$\begin{aligned}\widetilde{w}_\ell &= \sum_{k=1}^{\widetilde{N}} t_k \int_\Gamma \int_\Gamma \mathcal{Q}_x \mathcal{Q}_y k_{\tilde{p}}(x, y) \varphi_k(y) ds_y \psi_\ell(x) ds_x \\ &= \sum_{n=0}^{\tilde{p}} \int_\Gamma \mathcal{Q}_x f_n(x) \psi_\ell(x) ds_x \sum_{k=1}^{\widetilde{N}} t_k \int_\Gamma \mathcal{Q}_y g_n(y) \varphi_k(y) ds_y\end{aligned}$$

and the total effort would be reduced to $\mathcal{O}(\tilde{p}(\widetilde{N} + \widetilde{M}))$, as the coefficients

$$\widetilde{L}_n = \sum_{k=1}^{\widetilde{N}} t_k \int_\Gamma \mathcal{Q}_y g_n(y) \varphi_k(y) ds_y \qquad \text{for } n = 0, \ldots, \tilde{p}.$$

would be computed in $\mathcal{O}(\tilde{p}\widetilde{N})$ operations and the evaluation would take $\mathcal{O}(\tilde{p}\widetilde{M})$ operations.

But in general, the kernel approximation (10) is only valid for $|y| > d|x|$ with $d > 1$ and often an error estimate of the kind

$$|\mathcal{Q}_x \mathcal{Q}_y k(x, y) - \mathcal{Q}_x \mathcal{Q}_y k_{\tilde{p}}(x, y)| \leq c(\tilde{p}, d, |x|) \left(\frac{1}{d}\right)^{\tilde{p}+\varrho} \tag{11}$$

holds with some integer $\varrho \in \mathbb{Z}$. The constant $c(\tilde{p}, d, |x|)$ might be independent of $\tilde{p}$ or a polynomial in $\tilde{p}$ of low order. It also depends on d and $|x|$, but the error estimate is dominated by the exponential term $d^{-\tilde{p}-\varrho}$. Due to the restrictions

on the validation of the expansion, the matrix times vector multiplication is separated into two parts, the nearfield part and the farfield part. The farfield $\mathrm{FF}(\ell)$ is the set of indices k, for which the supports of the test function ψ_ℓ and the trial functions φ_k are well separated and therefore it is suitable to apply the kernel approximation (10) due to the error estimate (11). The nearfield part $\mathrm{NF}(\ell)$ of the matrix times vector multiplication is realized as in standard boundary element methods [38]. An exact definition of nearfield and farfield will be given later. Now, the matrix times vector multiplication reads as

$$\widetilde{w}_\ell = \sum_{k \in \mathrm{NF}(\ell)} A_h[\ell, k] t_k + \sum_{n=0}^{\tilde{p}} M_n(O, \psi_\ell) \widetilde{L}_n(0, \mathrm{FF}(\ell)). \tag{12}$$

The coefficients

$$M_n(O, \psi_\ell) = \int_\Gamma \mathcal{Q}_x f_n(x) \psi_\ell(x) ds_x,$$
$$L_n(O, \varphi_k) = \int_\Gamma \mathcal{Q}_y g_n(y) \varphi_k(y) ds_y,$$

with reference to a local center O, can either be computed exactly, for example, in the case of spherical harmonics [23, 24], or can be approximated by the use of some numerical quadrature rule. If the coefficients

$$\widetilde{L}_n(O, \mathrm{FF}(\ell)) = \sum_{k \in \mathrm{FF}(\ell)} t_k L_n(O, k) \qquad \text{for } n = 0, \ldots, \tilde{p} \tag{13}$$

are known an efficient realization of the matrix times vector multiplication will be given by (12). These coefficients depend on the vector $\underline{t}$. Therefore, they have to be recalculated in each matrix times vector multiplication. As the coefficients $\widetilde{L}_n(O, \mathrm{FF}(\ell))$ depend on the farfield of the support of the basis function ψ_ℓ, they differ from each other in general and an efficient calculation is necessary.

The efficient computation of the coefficients $\widetilde{L}_n(O, \mathrm{FF}(\ell))$ in (13) will be described only very briefly. More detailed descriptions can be found, e.g., in [7, 8]. In order to do this computation efficiently, as much information as possible is shared when these coefficients are determined.

The second basic idea of the fast multipole method, the hierarchical structure is applied to compute these expansions. First, this hierarchical structure, called cluster tree, has to be build based on geometrical information. This structure can either be based on the boundary elements or on the supports of the basis functions φ_k and ψ_ℓ. The realization of a boundary integral operator might differ for these two approaches, since the nearfields and farfields differ from each other. In the latter approach, two cluster trees have to be built if the trial and test functions do not coincide.

Here, we describe the construction of the cluster tree based on the supports of the basis functions φ_k and ψ_ℓ. In the case of using boundary elements τ_i

for the construction of the cluster structures, this construction is almost the same. Then only one cluster tree has to be built, but the setup of the nearfield part of the matrix and the evaluation of the farfield part of the matrix times vector multiplication might need some more effort for the assembling. Here, the cluster tree is built from the top down based on the supports of the basis functions φ_k and ψ_ℓ. All trial functions $\{\varphi_k\}_{k=1}^{\widetilde{N}}$ are included in a box containing the original domain Ω. The cluster ω_1^0 of level 0 consists of all trial functions $\{\varphi_k\}_{k=1}^{\widetilde{N}}$ or the corresponding set of indices. The hierarchical structure is build recursively by the refinement of the box corresponding to a cluster ω_i^λ of the level λ into eight similar boxes. The trial functions $\{\varphi_k\}_{k=1}^{\widetilde{N}}$ are assigned to the boxes due to the centers of their supports. All trial functions, which are assigned to one of the refined boxes, form the cluster $\omega_j^{\lambda+1}$ of the finer level $\lambda+1$ identified with the corresponding refined box. These clusters $\omega_j^{\lambda+1}$ are called the sons of the father cluster ω_i^λ. Empty boxes and the corresponding clusters containing no trial functions are neglected. This refinement is done until a minimal number of trial functions in the clusters is reached or until a maximal cluster level L is reached. Each of the trial functions φ_k is assigned to the cluster ω_i^L on the finest level L which contains the center of the support of φ_k. In this paper, we restrict our considerations to the case of a regular distribution of the boundary elements $\{\tau_k\}_{k=1}^{N}$ of a globally quasi uniform boundary element mesh. Nevertheless, the method can be extended to the adaptive case, see for example [2, 23].

Next, the second cluster tree with clusters σ_j^λ is build based on the supports of the test functions ψ_ℓ in the similar way. Depending on the choice of the two sets of basis functions, the two cluster trees might coincide.

We have used a more abstract definition of nearfield and farfield so far. Now, we can define these based on the cluster hierarchy. A cluster ω_i^λ is in the nearfield of a cluster σ_j^λ of the same level λ, if the condition

$$\operatorname{dist}\{C(\omega_i^\lambda), C(\sigma_j^\lambda)\} \le (d+1)\max\{r(\omega_i^\lambda), r(\sigma_j^\lambda)\} \tag{14}$$

holds for a parameter $d > 1$. $C(\omega_i^\lambda)$ denotes the center of the box identified with the cluster ω_i^λ, and $r(\omega_i^\lambda)$ is the corresponding radius of the cluster, i.e., $r(\omega_i^\lambda) = \sup_{x\in\omega_i^\lambda}|x - C(\omega_i^\lambda)|$. It is important for the multipole algorithm that the nearfield of a father cluster $\sigma_i^{\lambda-1}$ contains the nearfields of all its sons $\sigma_j^\lambda \subset \sigma_i^{\lambda-1}$. This definition of the nearfield and the farfield is transferred to the basis functions by their assignment to the leaves of the cluster tree:

$$\begin{aligned}\mathrm{NF}(\ell) := \Big\{ & k, 1 \le k \le \widetilde{N} \text{ and (14) holds for the cluster } \omega_i^L \text{ of } \varphi_k \\ & \text{and } \sigma_j^L \text{ is the cluster of } \psi_\ell. \Big\}, \\ \mathrm{FF}(\ell) := {} & \{1,\dots,N\} \setminus \mathrm{NF}(\ell).\end{aligned}$$

A symmetric definition of the nearfield helps to preserve the symmetry in the approximation of symmetric matrices, see for example [26, 30].

The efficient computation of the coefficients $\widetilde{L}_n(O, \mathrm{FF}(\ell))$ in (13) now uses this hierarchy. First, the coefficients

$$\widetilde{M}_n(C(\omega_j^L), \mathrm{P}(\omega_j^L)) = \sum_{k \in \omega_j^L} t_k \widehat{M}_n(C(\omega_j^L), \varphi_k) \tag{15}$$

are calculated for all clusters ω_j^L of the finest level L. $\mathrm{P}(\omega_j^\lambda) := \{k, \varphi_k \in \omega_j^\lambda\}$ is the set of all basis functions φ_k of the cluster ω_j^λ. The coefficients $\widehat{M}_n$ are given by

$$\widehat{M}_n(O, \varphi_k) := \int_\Gamma \mathcal{Q}_y f_n(y) \varphi_k(y) ds_y. \tag{16}$$

The coefficients $\widetilde{M}_n$ are now used to determine the multipole coefficients of the clusters on the coarser levels by a translation of the type

$$\widetilde{M}_n(C(\omega_j^\lambda), \mathrm{P}(\omega_j^\lambda)) = \sum_{\omega_i^{\lambda+1} \in \mathrm{sons}(\omega_j^\lambda)} \sum_s h_{n,s}^1(\overrightarrow{C(\omega_j^\lambda) C(\omega_i^{\lambda+1})}) \cdot \widetilde{M}_{n-s}(C(\omega_i^{\lambda+1}), \mathrm{P}(\omega_i^{\lambda+1})) \tag{17}$$

with some coefficients $h_{n,s}^1$. From these multipole coefficients of a cluster ω_j^λ, the needed local coefficients of a second cluster σ_i^λ in the farfield of ω_j^λ can be calculated by a conversion of the type

$$\widetilde{L}_n(C(\sigma_i^\lambda), \mathrm{P}(\omega_j^\lambda)) = \sum_s h_{n,s}^2(\overrightarrow{C(\omega_j^\lambda) C(\sigma_i^\lambda)}) \widetilde{M}_s(C(\omega_j^\lambda), \mathrm{P}(\omega_j^\lambda)) \tag{18}$$

with some coefficients $h_{n,s}^2$. These conversions are executed on the coarsest possible level, on which the admissibility condition (14) is fulfilled, i.e., for two clusters, which are in their mutual farfield, but their fathers are in their own mutual nearfields. These local coefficients are summed up for each cluster. Additionally, these coefficients are translated from each cluster σ_i^λ to its sons $\sigma_j^{\lambda+1}$ by

$$\widetilde{L}_n(C(\sigma_j^{\lambda+1}), \mathrm{FF}(\sigma_i^\lambda)) = \sum_s h_{n,s}^3(\overrightarrow{C(\sigma_i^\lambda) C(\sigma_j^{\lambda+1})}) \widetilde{L}_s(C(\sigma_i^\lambda), \mathrm{FF}(\sigma_i^\lambda)) \tag{19}$$

with some coefficients $h_{n,s}^3$. The sum of all coefficients $\widetilde{L}_n(C(\sigma_j^L), \cdot)$ results in the local coefficients $\widetilde{L}_n(C(\sigma_j^L), \mathrm{FF}(\ell))$ needed for the matrix times vector multiplication (12). Here, σ_j^L is the cluster to which the test function ψ_ℓ is assigned. Now, the coefficients $\widetilde{L}_n(C(\sigma_j^L), \mathrm{FF}(\ell))$ are known for a fast evaluation of the farfield part in the matrix times vector product (12). Note that all the translations and conversions have to be executed in each matrix times vector multiplication, as the coefficients in (15) change for each vector $\underline{t}$. As we have described the fast multipole method for an abstract kernel expansion, we have to require that the corresponding translations and conversions exist.

A fixed expansion degree $\tilde{p}$ is not sufficient to guarantee the asymptotic convergence rate of the fast boundary element method, in general. Instead, the expansion degree $\tilde{p}$ has to be chosen proportional to $\log^2 N$, as shown for example in [30] in the case of spherical harmonics. Therefore the total effort of one matrix times vector multiplication is of order $\mathcal{O}(N \log^2 N)$. The memory requirements are also of order $\mathcal{O}(N \log^2 N)$. We have described the fast multipole method in its original version as given by [7, 8]. Several approaches have been made since then to increase the performance of the method. Especially, the translations and the conversions of the multipole and local expansions have been optimized. For example, the effort for these operations can be reduced by fast Fourier transforms [5] or an exponential representation [9]. But this speedup really pays off for larger expansion degrees, which might usually not be necessary in the case of a fast boundary element method for the Laplace equation and for linear elastostatics. As long as the expansion degree $\tilde{p}$ has to be chosen proportional to $\log^2 N$, the total effort of the fast multipole method is not of order $\mathcal{O}(N)$ but higher, since always $\mathcal{O}(N)$ local expansions of $\tilde{p}+1$ coefficients have be evaluated.

A first approach to overcome this dependency of the expansion degree $\tilde{p}$ on the number of boundary elements has been made by [44] where the variable order approach of the panel clustering by [36] is transferred to the fast multipole method. In the case of boundary integral equations of the second kind with piecewise constant basis functions, one ends up with an $\mathcal{O}(N)$ algorithm. An approach that overcomes the dependency of the fast multipole method on the particular kernel expansion, which has to be derived for each differential operator separately, is given by [37].

4 Fast Boundary Element Methods for Linear Elastostatics

In this section, we try to show the differences of the approaches presented in [6], [29], and [46]. First, we consider the single layer potential of linear elastostatics

$$(Vt)(x) = \int_\Gamma U^*(x,y)t(y)ds_y \qquad \text{for } x \in \Gamma.$$

4.1 Realization of the Single Layer Potential as Linear Combination of the Kernel of the Laplacian and Its Derivatives

The fundamental solution $(U^*_{k\ell})_{\ell,k=1..3}$ of linear elastostatics

$$U^*_{k\ell}(x-y) = \frac{1+\nu}{8\pi E(1-\nu)}\left[(3-4\nu)\frac{\delta_{k\ell}}{|x-y|} + \frac{(x_k-y_k)(x_\ell-y_\ell)}{|x-y|^3}\right]$$

can be expressed by linear combinations of the kernel of the Laplacian and of its derivatives. In [6], the representation

$$U^*_{k\ell}(x-y) = \frac{1+\nu}{2E(1-\nu)}\frac{1}{4\pi}\left[(3-4\nu)\frac{\delta_{k\ell}}{|x-y|} - x_\ell\frac{\partial}{\partial x_k}\frac{1}{|x-y|} + \frac{\partial}{\partial x_k}\frac{y_\ell}{|x-y|}\right]$$

is chosen. A detailed analysis shows that the corresponding realization by the fast multipole methods requires four calls of the algorithm to compute the local expansions and seven evaluations of these local expansions in the case of a Galerkin matrix. But unfortunately, this realization of the Galerkin matrix V_h by the fast multipole method is not symmetric anymore, as a finite expansion degree has to be used.

4.2 Symmetric Realization of the Single Layer Potential as Linear Combination of the Kernel of the Laplacian and Its Derivatives

In the case of the Laplacian, the transposedness of the Galerkin matrices of the double layer potential and its adjoint operator can be preserved [26]. This gives the idea how to keep the symmetry of the Galerkin matrix of the single layer potential in linear elastostatics. The gradient terms are rewritten as

$$U^*_{k\ell}(x-y) = \frac{1+\nu}{2E(1-\nu)}\frac{1}{4\pi}\left[(3-4\nu)\frac{\delta_{k\ell}}{|x-y|} - x_\ell\frac{\partial}{\partial x_k}\frac{1}{|x-y|} - y_\ell\frac{\partial}{\partial y_k}\frac{1}{|x-y|}\right].$$

This representation preserves the symmetry within one block of the matrix. To guarantee the symmetry of the blocks, the expression of a block is added for interchanged indices k and ℓ and the sum is divided by two:

$$\begin{aligned} U^*_{k\ell}(x-y) = \frac{1+\nu}{2E(1-\nu)}\frac{1}{4\pi}\Bigg[& (3-4\nu)\frac{\delta_{k\ell}}{|x-y|} \\ & -\frac{1}{2}x_\ell\frac{\partial}{\partial x_k}\frac{1}{|x-y|} - \frac{1}{2}y_\ell\frac{\partial}{\partial y_k}\frac{1}{|x-y|} \\ & -\frac{1}{2}x_k\frac{\partial}{\partial x_\ell}\frac{1}{|x-y|} - \frac{1}{2}y_k\frac{\partial}{\partial y_\ell}\frac{1}{|x-y|}\Bigg]. \end{aligned} \tag{20}$$

The realization of the single layer potential by this representation requires six calls of the fast multipole algorithm to compute local coefficients and nine evaluations [27]. The number of evaluations can be reduced to six by a more involved implementation of the fast multipole algorithm, which needs to store more local coefficients. This representation leads to a more expensive application of the single layer potential, but the preserved symmetry of the Galerkin matrix is often advantageous for iterative solvers.

The Kernel Expansion for the Laplacian by Reformulated Spherical Harmonics

The separation of the variables in the kernel of the Laplacian is done by a expansion in spherical harmonics, in general. For a simpler implementation

and a fast realization, reformulated spherical harmonics [31, 45, 46] are used for the kernel expansion

$$\frac{1}{|x-y|} \approx \sum_{n=0}^{p} \sum_{m=-n}^{n} \overline{S_n^m}(y) R_n^m(x). \tag{21}$$

The reformulated spherical harmonics are given by

$$R_n^{\pm m}(x) = \frac{1}{(n+m)!} \frac{d^m}{du^m} P_n(u)\big|_{u=\widehat{x}_3} (\widehat{x}_1 \pm i\widehat{x}_2)^m |x|^n,$$
$$S_n^{\pm m}(y) = (n-m)! \frac{d^m}{du^m} P_n(u)\big|_{u=\widehat{y}_3} (\widehat{y}_1 \pm i\widehat{y}_2)^m \frac{1}{|y|^{n+1}}$$

in Cartesian coordinates for $m \geq 0$ and $\widehat{y}_i = y_i/|y|$. They can be computed efficiently by recursive schemes. $P_n(u)$ denote the Legendre polynomials. In the case of this expansion, the multipole coefficients (16) are computed by

$$\widehat{M}_n^m(O, \varphi_k) := \int_\Gamma \mathcal{Q}_y R_n^m(y) \varphi_k(y) ds_y$$

and form the multipole coefficients (15) of a cluster ω_j^L by

$$\widetilde{M}_n^m(C(\omega_j^L), \mathrm{P}(\omega_j^L)) = \sum_{k \in \omega_j^L} t_k \widehat{M}_n^m(C(\omega_j^L), \varphi_k).$$

The translation (17) of multipole coefficients reads as

$$\widetilde{M}_n^m(C(\omega_j^\lambda), \mathrm{P}(\omega_j^\lambda)) = \sum_{\omega_i^{\lambda+1} \in \mathrm{sons}(\omega_j^\lambda)} \sum_{s=0}^{n} \sum_{t=-s}^{s} R_s^t(\overrightarrow{C(\omega_j^\lambda) C(\omega_i^{\lambda+1})}) \cdot \widetilde{M}_{n-s}^{m-t}(C(\omega_i^{\lambda+1}), \mathrm{P}(\omega_i^{\lambda+1})).$$

The conversion (18) of multipole coefficients to local coefficients takes the form

$$\widetilde{L}_n^m(C(\sigma_i^\lambda), \mathrm{P}(\omega_j^\lambda)) = \sum_{s=0}^{\infty} \sum_{t=-s}^{s} (-1)^n \overline{S_{n+s}^{m+t}}(\overrightarrow{C(\omega_j^\lambda) C(\sigma_i^\lambda)}) \widetilde{M}_s^t(C(\omega_j^\lambda), \mathrm{P}(\omega_j^\lambda)),$$

while the translation (19) of local coefficients is executed by

$$\widetilde{L}_n^m(C(\sigma_j^{\lambda+1}), \mathrm{FF}(\sigma_i^\lambda)) = \sum_{s=n}^{p} \sum_{t=-s}^{s} R_{s-n}^{t-m}(\overrightarrow{C(\sigma_i^\lambda) C(\sigma_j^{\lambda+1})}) \widetilde{L}_s^t(C(\sigma_i^\lambda), \mathrm{FF}(\sigma_i^\lambda)).$$

With these conversion and translation formulae, all ingredients of the fast multipole method for the kernel of the Laplacian are now given. So the fast multipole method can be applied to the kernel of the Laplacian and to linear combinations of derivatives of this kernel. Also the single layer potential of linear elastostatics can be realized by these means, as described before in the approaches of [6] and [29].

4.3 Realization of the Boundary Integral Operators by an Expansion of the Kernel of Linear Elastostatics

In the approach of [46], the expansion of

$$|x-y| = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(\frac{\overline{S_n^m}(y)|x|^2 R_n^m(x)}{2n+3} - \frac{|y|^2 \overline{S_n^m}(y) R_n^m(x)}{2n-1} \right),$$

for $|x| < |y|$, is used to derive an expansion of the fundamental solution in linear elastostatics,

$$U_{k\ell}^*(x-y) = \frac{1}{8\pi\mu} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(F_{k\ell,n}^m(x) \overline{S_n^m}(y) + G_{k,n}^m(x) \overline{S_n^m}(y) y_\ell \right). \tag{22}$$

The coefficients $F_{k\ell,n}^m(x)$ and $G_{k,n}^m(x)$ are defined by

$$F_{k\ell,n}^m(x) = \frac{\lambda+3\mu}{\lambda+2\mu} \delta_{k\ell} R_n^m(x) - \frac{\lambda+\mu}{\lambda+2\mu} x_\ell \frac{\partial}{\partial x_k} R_n^m(x),$$
$$G_{k,n}^m(x) = \frac{\lambda+\mu}{\lambda+2\mu} \frac{\partial}{\partial x_k} R_n^m(x).$$

This expansion is used in [46] to realize the boundary integral operators of linear elastostatics. For the single layer potential, the matrix times vector product takes the form

$$\widetilde{w}_{\ell,i} = \sum_{j=1}^{3} \sum_{k \in \mathrm{NF}(\ell)} V_h^{ij}[\ell,k] t_{k,j} + \frac{1}{8\pi\mu} \sum_{n=0}^{p} \sum_{m=-n}^{n} \left(\sum_{j=1}^{3} F_{ij,n}^m(x) \widetilde{L}_{j,n}^{1,m}(\mathrm{FF}(\ell)) + G_{i,n}^m(x) \widetilde{L}_n^{2,m}(\mathrm{FF}(\ell)) \right). \tag{23}$$

$\widetilde{w}_{\ell,i}$ denotes the ℓ–th entry of the vector $\underline{\widetilde{w}}$ in the i–th coordinate. The computation of the local coefficients $\widetilde{L}_{j,n}^{1,m}(\mathrm{FF}(\ell))$ and $\widetilde{L}_n^{2,m}(\mathrm{FF}(\ell))$ is described next. The multipole coefficients

$$\widehat{M}_{k,n}^{1,m}(O,\varphi_i) = \int_\Gamma R_n^m(y) \varphi_{i,k}(y) ds_y \tag{24}$$

$$\widehat{M}_{k,n}^{2,m}(O,\varphi_i) = \int_\Gamma R_n^m(y) y_k \varphi_{i,k}(y) ds_y \tag{25}$$

have to be computed for each basis functions and for $k = 1, \ldots, 3$, similar to (16). $\varphi_{i,k}$ denotes the k–th component of the trial function φ_i. These coefficients can be computed once in advance and be reused in each matrix times vector multiplication. Due to the more involved expansion series, four sets of coefficients have to be computed in this approach. In each matrix times vector multiplication the coefficients

$$\widetilde{M}_{k,n}^{1,m}(C(\omega_j^L),\mathrm{P}(\omega_j^L)) = \sum_{i\in\omega_j^L} t_{i,k}\widehat{M}_{k,n}^{1,m}(C(\omega_j^L),\varphi_i)$$

$$\widetilde{M}_{n}^{2,m}(C(\omega_j^L),\mathrm{P}(\omega_j^L)) = \sum_{i\in\omega_j^L}\sum_{\ell=1}^{3} t_{i,\ell}\widehat{M}_{\ell,n}^{2,m}(C(\omega_j^L),\varphi_i).$$

are calculated for all clusters ω_j^L of the finest level first as in (15). $t_{i,k}$ denotes the i–th entry of the vector $\underline{t}$ for the k–th component. The translation (17) of these multipole coefficients now looks like

$$\widetilde{M}_{k,n}^{1,m}(C(\omega_j^\lambda),\mathrm{P}(\omega_j^\lambda)) = \sum_{\omega_i^{\lambda+1}\in\mathrm{sons}(\omega_j^\lambda)}\sum_{s=0}^{n}\sum_{t=-s}^{s} R_s^t(z)\widetilde{M}_{k,n-s}^{1,m-t}(C(\omega_i^{\lambda+1}),\mathrm{P}(\omega_i^{\lambda+1})),$$

$$\widetilde{M}_{n}^{2,m}(C(\omega_j^\lambda),\mathrm{P}(\omega_j^\lambda)) = \sum_{\omega_i^{\lambda+1}\in\mathrm{sons}(\omega_j^\lambda)}\sum_{s=0}^{n}\sum_{t=-s}^{s} R_s^t(z)\Bigg(\widetilde{M}_{n-s}^{2,m-t}(C(\omega_i^{\lambda+1}),\mathrm{P}(\omega_i^{\lambda+1})) + \sum_{\ell=1}^{3} z_\ell\widetilde{M}_{\ell,n-s}^{1,m-t}(C(\omega_i^{\lambda+1}),\mathrm{P}(\omega_i^{\lambda+1}))\Bigg)$$

where $z = \overrightarrow{C(\omega_j^\lambda)C(\omega_i^{\lambda+1})}$. The conversion (18) of multipole coefficients to local coefficients takes the form

$$\widetilde{L}_{\ell,n}^{1,m}(C(\sigma_i^\lambda),\mathrm{P}(\omega_j^\lambda)) = \sum_{s=0}^{\infty}\sum_{t=-s}^{s}(-1)^n\overline{S_{n+s}^{m+t}}(z)\widetilde{M}_{\ell,s}^{1,t}(C(\omega_j^\lambda),\mathrm{P}(\omega_j^\lambda)),$$

$$\widetilde{L}_{n}^{2,m}(C(\sigma_i^\lambda),\mathrm{P}(\omega_j^\lambda)) = \sum_{s=0}^{\infty}\sum_{t=-s}^{s}(-1)^n\overline{S_{n+s}^{m+t}}(z)\Bigg(\widetilde{M}_{s}^{2,t}(C(\omega_j^\lambda),\mathrm{P}(\omega_j^\lambda)) - \sum_{\ell=1}^{3} z_\ell\widetilde{M}_{\ell,s}^{1,t}(C(\omega_j^\lambda),\mathrm{P}(\omega_j^\lambda))\Bigg),$$

where $z = \overrightarrow{C(\omega_j^\lambda)C(\sigma_i^\lambda)}$, while the translation (19) of local coefficients is evaluated by

$$\widetilde{L}_{\ell,n}^{1,m}(C(\sigma_j^{\lambda+1}),\mathrm{FF}(\sigma_i^\lambda)) = \sum_{s=n}^{p}\sum_{t=-s}^{s} R_{s-n}^{t-m}(z)\widetilde{L}_{\ell,s}^{1,t}(C(\sigma_i^\lambda),\mathrm{FF}(\sigma_i^\lambda)),$$

$$\widetilde{L}_{n}^{2,m}(C(\sigma_j^{\lambda+1}),\mathrm{FF}(\sigma_i^\lambda)) = \sum_{s=n}^{p}\sum_{t=-s}^{s} R_{s-n}^{t-m}(z)\Bigg(\widetilde{L}_{s}^{2,t}(C(\sigma_i^\lambda),\mathrm{FF}(\sigma_i^\lambda)) - \sum_{\ell=1}^{3} z_\ell\widetilde{L}_{\ell,s}^{1,t}(C(\sigma_i^\lambda),\mathrm{FF}(\sigma_i^\lambda))\Bigg)$$

where $z = \overrightarrow{C(\sigma_i^\lambda)C(\sigma_j^{\lambda+1})}$.

In the case of the double layer potential K, its adjoint operator K' and the hypersingular operator D, operators $\mathcal{Q}_x$ and $\mathcal{Q}_y$, which are linear combinations of derivatives with respect to x and y, have to be applied to this expansion. In detail, the operator $\mathcal{Q}_y$ is applied in the computation of the multipole coefficients in (24) and (25), and the operator $\mathcal{Q}_x$ is applied in the evaluation of the expansion (23). Therefore, all boundary integral operators require the computation of four sets of coefficients $\widetilde{L}_{\ell,n}^{1,m}$ and $\widetilde{L}_{n}^{2,m}$. This corresponds to four calls of the fast multipole method, but the translations, the conversions and the evaluations get a little more involved.

4.4 Realization of the Double Layer Potential as Linear Combination of Derivatives of the Kernel of the Laplacian

In the approach of [6], the kernel $T^*(x,y) = (\gamma_{1,y} U^*(x,y))^\top$ of the double layer potential K is rewritten, in a similar way as the kernel of the single layer potential, as

$$T^*_{k\ell}(x,y) = \sum_{j=1}^{3} R_{k\ell j} \left[n_j(y) \frac{1}{|x-y|} \right] - \sum_{j=1}^{3} \frac{1}{8\pi(1-\nu)} \frac{\partial}{\partial x_k} \frac{\partial}{\partial x_j} \left[n_j(y) y_\ell \frac{1}{|x-y|} \right]$$

with an operator

$$R_{k\ell j} = \frac{1}{8\pi(1-\nu)} \left[(1-2\nu) \left(\delta_{\ell j} \frac{\partial}{\partial x_k} - \delta_{k\ell} \frac{\partial}{\partial x_j} \right) - 2(1-\nu) \delta_{kj} \frac{\partial}{\partial x_\ell} + x_\ell \frac{\partial}{\partial x_k} \frac{\partial}{\partial x_j} \right].$$

The realization of this representation by the fast multipole method ends up with twelve calls of the fast multipole algorithm of the Laplacian. Therefore, the effort of a application of the double layer potential K is more expensive than the corresponding realization of the single layer potential.

4.5 Realization of Boundary Integral Operators using Integration by Parts

In our approach [29], we use a representation of the double layer potential K of linear elastostatics by weakly singular boundary integral operators which can be derived by integration by parts [18]. The double layer potential K can be rewritten by

$$\begin{aligned} (Ku)(x) = \frac{1}{4\pi} \int_\Gamma u(y) \frac{\partial}{\partial n_y} \frac{1}{|x-y|} ds_y - \frac{1}{4\pi} \int_\Gamma \frac{1}{|x-y|} (\mathcal{M}u)(y) ds_y \\ + 2\mu \left(V_E (\mathcal{M}u) \right)(x). \end{aligned} \tag{26}$$

with $\mu = E/(2(1+\nu))$ and an operator $\mathcal{M}$, consisting of components of the surface curl,

$$\mathcal{M} = \begin{pmatrix} 0 & n_2\frac{\partial}{\partial x_1} - n_1\frac{\partial}{\partial x_2} & n_3\frac{\partial}{\partial x_1} - n_1\frac{\partial}{\partial x_3} \\ n_1\frac{\partial}{\partial x_2} - n_2\frac{\partial}{\partial x_1} & 0 & n_3\frac{\partial}{\partial x_2} - n_2\frac{\partial}{\partial x_3} \\ n_1\frac{\partial}{\partial x_3} - n_3\frac{\partial}{\partial x_1} & n_2\frac{\partial}{\partial x_3} - n_3\frac{\partial}{\partial x_2} & 0 \end{pmatrix}.$$

The representation (26) allows to realize the double layer potential of linear elastostatics by six calls of the fast multipole method and nine evaluations [27]. In a more involved implementation, this can be reduced to six evaluations again. Therefore the effort for an application of the double layer potential is comparable to the application of our realization of the single layer potential and not increased as in the approach of [6].

The representation (26) of the double layer potential is also used in the nearfield. Therefore, only weakly singular boundary integral operators have be be evaluated. The effort for the computation of the Galerkin weights is significantly reduced by the change from Cauchy singular to weakly singular integrals.

In the case of piecewise linear trial functions and plane triangles as boundary elements, $\mathcal{M}$ maps the linear basis functions to piecewise constant basis functions. Therefore, the already computed nearfield matrices of the single layer potentials can be reused.

The representation (26) of the double layer potential can be used to rewrite the bilinear form of its adjoint operator K' as

$$\langle K't, v\rangle_\Gamma = \langle K'_L t, v\rangle_\Gamma - \langle V_L t, \mathcal{M}v\rangle_\Gamma + 2\mu\langle Vt, \mathcal{M}v\rangle_\Gamma. \tag{27}$$

Here, V_L and K'_L denote the corresponding operators of the Laplacian which are applied componentwise. This is sufficient for the used Galerkin method. In this way, the bilinear form of the adjoint double layer potential K' can be realized by six calls of the fast multipole algorithm of the Laplacian.

Using piecewise constant trial functions and piecewise linear, continuous test functions, the already computed nearfield matrices can be reused for the realization of the bilinear form of the adjoint double layer potential.

As in the case of the Laplace equation, the bilinear form of the hypersingular operator can be transformed to bilinear forms of single layer potentials. Based on the representation (26) of the double layer potential K, integration by parts reduces the bilinear form of the hypersingular operator to [13]

$$\begin{aligned}\langle Du, v\rangle_\Gamma = &\int_\Gamma\int_\Gamma \frac{\mu}{4\pi}\frac{1}{|x-y|}\left(\sum_{k=1}^{3}\left(\mathcal{M}_{k+2,k+1}v\right)(x)\cdot\left(\mathcal{M}_{k+2,k+1}u\right)(y)\right)ds_y ds_x \\ &+\int_\Gamma\int_\Gamma (\mathcal{M}v)^\top(x)\left(\frac{\mu}{2\pi}\frac{I}{|x-y|} - 4\mu^2 U^*(x,y)\right)(\mathcal{M}u)(y)ds_y ds_x \\ &+\int_\Gamma\int_\Gamma \sum_{i,j,k=1}^{3}\left(\mathcal{M}_{k,j}v_i\right)(x)\frac{\mu}{4\pi}\frac{1}{|x-y|}\left(\mathcal{M}_{k,i}u_j\right)(y)ds_y ds_x.\end{aligned}$$

In the first line the indices 4 and 5 of the operator $\mathcal{M}$ have to be identified with 1 and 2, respectively.

Overall, it is sufficient to have a fast realization of the single layer potentials and the double layer potential of the Laplacian in our approach. Therefore, the effort for the computation of the Galerkin weights is reduced significantly. The approach based on integration by parts is not restricted to the fast multipole method but can also be used for other fast boundary element techniques.

By a detailed analysis [27], we have shown that the use of the fast multipole method as a fast boundary element method does not effect the main properties and the asymptotic error estimate of the boundary element method, summarized in the following theorem.

Theorem 1 ([27]). *Let $\widehat{t} \in \widetilde{H}^{\sigma}_{pw}(\Gamma_D)$ and $\widehat{u} \in \widetilde{H}^{\eta}(\Gamma_N)$ for $\sigma \in [0,1]$ and $\eta \in [1,2]$ be the unique solution of the variational problem (5). Let the discretization of the boundary be shape–regular and quasi–uniform. Let the expansion degree p of the multipole expansion (21) be proportional to $\log N$. Then the variational problem of the approximations of the boundary integral operators by the fast multipole method is uniquely solvable. For the approximate solutions $\widetilde{u}_h$ and $\widetilde{t}_h$ the following error estimate holds:*

$$\|\widehat{t} - \widetilde{t}_h\|^2_{H^{-1/2}(\Gamma)} + \|\widehat{u} - \widetilde{u}_h\|^2_{H^{1/2}(\Gamma)} \leq c \left(h^{2\sigma+1} \|\widehat{t}\|^2_{H^{\sigma}_{pw}(\Gamma)} + h^{2\eta-1} \|\widehat{u}\|^2_{H^{\eta}(\Gamma)} \right).$$

Similar results should hold for the other approaches to realize the boundary integral operators by a fast multipole method, as all approaches are based on the expansion in spherical harmonics. It seems to be an open question, how the expansion degrees have to be chosen optimally in each of the approaches. This optimal choice of the expansion degree has a big influence on the performance of the methods and has to be considered in comparisons between the methods.

As a fixed expansion degree is not sufficient to keep up with the asymptotic error estimate of a boundary element method, the expansion degree has to be adopted to the number of boundary elements like $\log N$. With a fixed expansion degree p over all levels in the cluster tree of the fast multipole method, the effort of the fast multipole method is always of order $\mathcal{O}(N \log^2 N)$, as for each boundary element an expansion with $\mathcal{O}(\log^2 N)$ coefficients has to be evaluated.

To overcome this logarithmic terms in the effort, an attempt has been made by [44] where the variable order approach of the panel clustering [36] is transferred to the fast multipole method. In the case of boundary integral equations of the second kind with piecewise constant basis functions, one ends up with an $\mathcal{O}(N)$ algorithm.

5 Numerical Examples

Finally, we show first some academic examples and then some examples of industrial interest. First, we compare our version [29] of the fast multipole

method for linear elastostatics with a standard boundary element approach. The considered domain is the cuboid shown in Fig. 1.

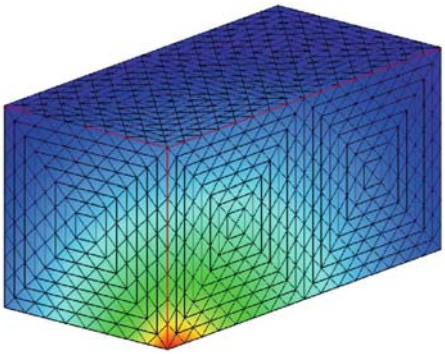

Fig. 1. Cuboid with 2560 boundary elements.

The left front side is the part with Dirichlet boundary conditions. The rest of the boundary has Neumann boundary conditions. The given boundary data are the traces of a chosen solution of the boundary value problem (1), which is given by a fundamental solution with the singularity outside of the domain. The system of linear equations (8) is solved as Schur complement system, with an iterative inversion of the matrix of the single layer potential in each iteration step. The results of this computations are given in Table 1.

Table 1. Comparison of standard and fast BEM.

L	N	M	dof	t_1	t_2	it	D-error	N-error
0	40	22	63	0	0	19	1.25e-3	7.23e-2
				0	0	19	1.25e-3	7.23e-2
1	160	82	255	2	0	24	2.86e-4	4.71e-2
				3	2	24	2.86e-4	4.71e-2
2	640	322	1035	37	5	27	5.15e-5	2.26e-2
				17	18	27	5.14e-5	2.26e-2
3	2560	1282	4179	609	82	30	1.06e-5	1.05e-2
				111	150	30	1.05e-5	1.05e-2
4	10240	5122	16803	(162 min)	(>23 min)			
				7.6 min	13.5 min	32	2.47e-6	5.06e-3
5	40960	20482	67395	(43.3 h)	(>6.6 h)			
				0.5 h	1.3 h	34	5.96e-7	2.50e-3

The first line of each refinement level L shows the data for the standard boundary element method, while the data in the second line refer to the fast boundary element method. N is the number of triangles and M denotes the number of nodes. Further, the number dof of degrees of freedom is given. t_1 and t_2 are the times in seconds for setting up and for solving the system of linear equations. It is the number of iteration steps needed in the Schur complement conjugate gradient method with a relative accuracy of 10^{-8}. Further, the errors of the computed Dirichlet and Neumann data are compared using the $L_2(\Gamma)$ norm. These experiments have been carried out on personal computer with an Intel Pentium 4 processor with 3.06 GHz and 1 GB of RAM. An artificial multilevel boundary element preconditioner [41] and the technique of operators of opposite order [28, 43] have been used as preconditioners for the single layer potential and the hypersingular operator, respectively.

The numbers of iterations are the same for the standard and the fast boundary element method. This is an evidence that the approximation of the system matrix is rather good and that important properties such as the symmetry are preserved. The standard method only works until the fourth refinement level due to the memory restrictions. Therefore, some values have been extrapolated and have been put in brackets. The numbers of iterations grow logarithmically as expected from theory [22]. There is some overhead in the fast boundary element method on the first refinement levels. But the fast multipole method speeds up the calculations on the higher levels significantly. The speedup is larger for setting up the system than for solving. This is typically for the fast multipole method. But here it is also caused by the special choice of the parameters of the fast multipole method for an optimal total time and by some implementation details. The factors of the increasing computational times for solving the Schur complement system are a little bit higher than expected for an algorithm of $\mathcal{O}(N \log^2 N)$. This is due to the increasing number of iterations in the inner iteration for the inversion of the single layer potential in the Schur complement conjugate gradient method. The errors of the Dirichlet and Neumann data match each other very well for the standard and the fast boundary element method. The convergence rates expected from theory are obtained.

In Table 2, a diagonal scaling, the artificial multilevel boundary element preconditioner [41] and an algebraic multigrid method [19, 20, 27] for the fast multipole method are compared as preconditioners of the single layer potential in a Dirichlet boundary value problem for the cuboid in Fig. 1. In the latter case, the algebraic multigrid preconditioner of the single layer potential of the Laplacian is applied componentwise, since its application is cheaper than the application of the operator of linear elastostatics and gives good iteration numbers.

Overall, six uniform refinement steps have been applied such that the finest grid consists of 163840 triangles and 491520 degrees of freedom. The computations have been executed on a personal computer with an AMD Opteron processor 146 with 2.0 GHz and 4 GB RAM. The iteration numbers of the

Table 2. Comparison of the preconditioners.

			scaling			ABPX			AMG		
L	N	dof	t_1	t_2	It	t_1	t_2	It	t_1	t_2	It
0	40	120	0	0	26	0	0	26	0	0	13
1	160	480	1	2	36	2	1	33	2	1	15
2	640	1920	5	13	51	5	10	40	5	4	16
3	2560	7680	18	93	70	19	58	44	21	25	16
4	10240	30720	75	680	92	76	370	50	88	160	17
5	40960	122880	365	6945	124	368	3080	55	457	1392	19
6	163840	491520	1749	55984	165	1750	20386	60	2325	9481	21

diagonal scaling grow quite fast. The iteration numbers of the artificial multilevel preconditioner increase logarithmically as predicted. As the costs for this preconditioner are very low, the reduced number of iterations results in a faster solving of the system. The algebraic multigrid method reduces the number of iterations once more. The application of the algebraic multigrid preconditioner is a lot more expensive than the artificial multilevel preconditioner, but nevertheless the computational times are reduced again. Therefore, the extra effort to set up the algebraic multilevel preconditioner is justifiable.

The first example of industrial interest is the stress analysis for a part of a press equipment and has been provided by W. Volk, M. Wagner and S. Wittig (BMW Research Center Munich). The two pictures in Fig. 2 show the deformed body under imposed deformations and stresses.

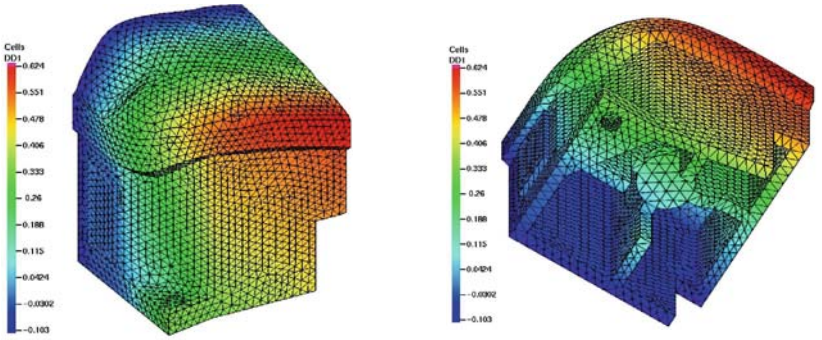

Fig. 2. Part of a press equipment.

The numbers N of boundary elements, the numbers of degrees of freedoms, the computational times for setting up and solving the system of linear equations and the numbers of iterations of the conjugate gradient method with a relative accuracy of 10^{-8} are given in Table 3. The press equipment is only fixed at a few points. Therefore, the block of the single layer potential is set

up completely and inverted by a direct method. Than the complete system is solved as a Schur complement system on the Opteron computer.

Table 3. Computational times for the press equipment.

N	DOF(N)	DOF(D)	Setup	Solving	it
13144	182	19586	896	3398	343
52572	728	78426	3940	23623	372

The numbers of iterations might seem to be rather high on the first sight. But they are caused by the thin walls of the body. The slight increase of the numbers of iterations shows that the preconditioning of the hypersingular operator by the operator of opposite order [43] performs as expected.

The second example is a metal foam, see Fig. 3, provided by H. Andrä (Fraunhofer–Institut für Techno– und Wirtschaftsmathematik, Kaiserslautern). The left picture shows the body in the reference configuration, while the right picture shows the deformed body. The bottom side of the foam has been fixed and the top side has been pressed down by a given deformation in z–direction.

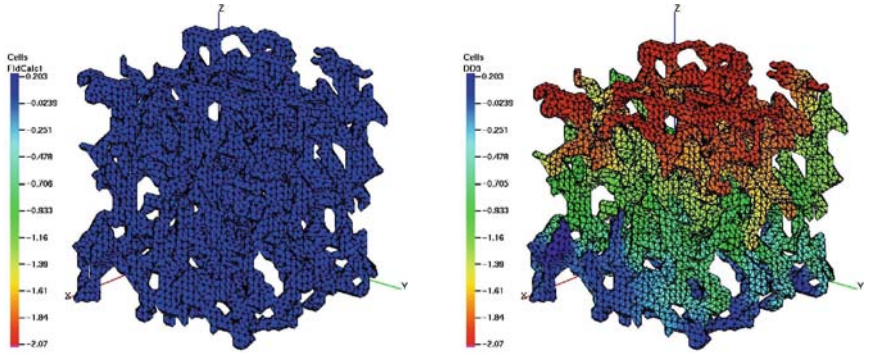

Fig. 3. Undeformed and deformed foam of metal.

The number of boundary elements, the number of nodes, the number of degrees of freedom, the number of iterations and the computational times are given in Table 4.

Table 4. Computational times for the foam.

N	M	DOF(N)	DOF(D)	Setup	Solving	It
28952	14152	396	41511	1730	9832	264

Several attempts of computations with commercial finite element software had been failed for this complex structure. But these computations were possible with the fast boundary element method. The preconditioners work well for this complex structure, too. The computations are rather costly for fast boundary element methods, as the boundary element mesh fills out the whole volume and the nearfields in the cluster tree are very large, consequently.

Overall, the realization of the boundary element method for linear elastostatics by the fast multipole method works very well and is applicable to complex structures of industrial interest.

Acknowledgment

This work has been supported by the German Research Foundation 'Deutsche Forschungsgemeinschaft (DFG)' under the grant SFB 404 'Multifield Problems in Continuum Mechanics' at the University of Stuttgart.

References

1. M. Bebendorf, S. Rjasanow: Adaptive low-rank approximation of collocation matrices. Computing 70 (2003) 1–24.
2. H. Cheng, L. Greengard, V. Rokhlin: A fast adaptive multipole algorithm in three dimensions. J. Comput. Phys. 155 (1999) 468–498.
3. M. Costabel: Symmetric methods for the coupling of finite elements and boundary elements. In: Boundary elements IX, Vol. 1, pp. 411–420. Comput. Mech., Southampton, 1987.
4. W. Dahmen, S. Prößdorf, R. Schneider: Wavelet approximation methods for pseudodifferential equations. II: Matrix compression and fast solution. Adv. Comput. Math. 1 (1993) 259–335.
5. W. D. Elliott, J. A. Board: Fast Fourier transform accelerated fast multipole algorithm. SIAM J. Sci. Comput. 17 (1996) 398–415.
6. Y. Fu, K. J. Klimkowski, G. J. Rodin, E. Berger, J. C. Browne, J. K. Singer, R. A. van de Geijn, K. S. Vemaganti: A fast solution method for three-dimensional many-particle problems of linear elasticity. Internat. J. Numer. Methods Engrg. 42 (1998) 1215–1229.
7. L. Greengard: The Rapid Evaluation of Potential Fields in Particle Systems. The MIT Press, Cambridge, MA, 1987.
8. L. Greengard, V. Rokhlin: A fast algorithm for particle simulations. J. Comput. Phys. 73 (1987) 325–348.
9. L. Greengard, V. Rokhlin: A new version of the fast multipole method for the Laplace equation in three dimensions. Acta Numerica 6 (1997) 229–269.
10. W. Hackbusch: A sparse matrix arithmetic based on $\mathcal{H}$-matrices. I: Introduction to $\mathcal{H}$-matrices. Computing 62 (1999) 89–108.

11. W. Hackbusch, B. Khoromskij, S. A. Sauter: On $\mathcal{H}^2$-matrices. In: Lectures on applied mathematics (H.-J. Bungartz et al. eds.), Proceedings of the symposium organized by the Sonderforschungsbereich 438 on the occasion of Karl-Heinz Hoffmann's 60th birthday, Munich, Germany, June 30-July 1, 1999, pp. 9–29, Springer, Berlin, 2000.
12. W. Hackbusch, Z. P. Nowak: On the fast matrix multiplication in the boundary element method by panel clustering. Numer. Math. 54 (1989) 463–491.
13. H. Han: The boundary integro-differential equations of three-dimensional Neumann problem in linear elasticity. Numer. Math. 68 (1994) 269–281.
14. K. Hayami, S. A. Sauter: A panel clustering method for 3-D elastostatics using spherical harmonics. In: Integral Methods in Science and Engineering (B. Bertram et al. eds.), Proceedings of the 5th International Conference, IMSE'98, Chapman Hall/CRC Res. Notes Math., Vol. 418, pp. 179–184, Chapman & Hall/CRC, Boca Raton, 2000.
15. G. C. Hsiao, W. L. Wendland: On a boundary integral method for some exterior problems in elasticity. Mat. Mech. Astron. 257 (1985) 31–60.
16. G. C. Hsiao, W. L. Wendland: Boundary element methods: foundation and error analysis. In: Encyclopedia of Computational Mechanics, Vol. I: Fundamentals (E. Stein, R. de Borst, T. Hughes eds.), pp. 339–373, John Wiley & Sons, 2004.
17. G. C. Hsiao, W. L. Wendland: Boundary Integral Equations: Variational Methods. Springer, Heidelberg, to be published.
18. V. D. Kupradze, T. G. Gegelia, M. O. Baseleisvili, T. V. Burculadze: Three-dimensional problems of the mathematical theory of elasticity and thermoelasticity. North-Holland Series in applied Mathematics and Mechanics, Vol. 25, North-Holland Publishing Company, Amsterdam, New York, Oxford, 1979.
19. U. Langer, D. Pusch: Data-sparse algebraic multigrid methods for large scale boundary element equations. Appl. Numer. Math. 54 (2005) 406–424.
20. U. Langer, D. Pusch, S. Reitzinger: Efficient preconditioners for boundary element matrices based on grey-box algebraic multigrid methods. Internat. J. Numer. Methods Engrg. 58 (2003) 1937–1953.
21. W. McLean: Strongly elliptic systems and boundary integral equations. Cambridge University Press, Cambridge, 2000.
22. W. McLean, O. Steinbach: Boundary element preconditioners for a hypersingular integral equation on an interval. Adv. Comput. Math. 11 (1999) 271–286.
23. K. Nabors, F. T. Korsmeyer, F. T. Leighton, J. White: Preconditioned, adaptive, multipole-accelerated iterative methods for three-dimensional first-kind integral equations of potential theory. SIAM J. Sci. Comput. 15 (1994) 713–735.
24. J. N. Newman: Distributions of sources and normal dipoles over a quadrilateral panel. J. Engrg. Math. 20 (1986) 113–126.
25. N. Nishimura: Fast multipole accelerated boundary integral equation methods. Appl. Mech. Rev. 55 (2002) 299–324.
26. G. Of: Die Multipolmethode für Randintegralgleichungen. Diplomarbeit, Universität Stuttgart, 2001.
27. G. Of: BETI–Gebietszerlegungsmethoden mit schnellen Randelementverfahren und Anwendungen. Doctoral thesis, Universtät Stuttgart, 2006.
28. G. Of, O. Steinbach: A fast multipole boundary element method for a modified hypersingular boundary integral equation. In: Analysis and simulation of multifield problems (W. L. Wendland et al. eds.), Selected papers of the international conference on multifield problems, Stuttgart, Germany, April 8-10, 2002. Lect. Notes Appl. Comput. Mech., Vol. 12, pp. 163–169, Springer, Berlin, 2003.

29. G. Of, O. Steinbach, W. L. Wendland: Applications of a fast multipole Galerkin boundary element method in linear elastostatics. Comput. Visual. Sci. 8 (2005) 201–209.
30. G. Of, O. Steinbach, W. L. Wendland: The fast multipole method for the symmetric boundary integral formulation. IMA J. Numer. Anal. 26 (2006) 272–296.
31. J. M. Perez-Jorda, W. Yang: A concise redefinition of the solid spherical harmonics and its use in the fast multipole methods. J. Chem. Phys. 104 (1996) 8003–8006.
32. V. Popov, H. Power: An $O(N)$ Taylor series multipole boundary element method for three-dimensional elasticity problems. Eng. Anal. Bound. Elem. 25 (2001) 7–18.
33. S. Rjasanow, O. Steinbach: The Fast Solution of Boundary Integral Equations. Springer Series in Mathematical and Analytical Technology with Applications to Engineering, Vol. 13, Springer, New York, 2006.
34. V. Rokhlin: Rapid solution of integral equations of classical potential theory. J. Comput. Phys. 60 (1985) 187–207.
35. S. A. Sauter, C. Schwab: Randelementmethoden. Analyse, Numerik und Implementierung schneller Algorithmen. B.G. Teubner, Stuttgart, Leipzig, Wiesbaden, 2004.
36. S. A. Sauter: Variable order panel clustering. Computing 64 (2000) 223–261.
37. G. Schmidlin, C. Lage, C. Schwab: Rapid solution of first kind boundary integral equations in $\mathbb{R}^3$. Eng. Anal. Bound. Elem. 27 (2003) 469–490.
38. C. Schwab, W. L. Wendland: On numerical cubatures of singular surface integrals in boundary element methods. Numer. Math. 62 (1992) 343–369.
39. S. Sirtori: General stress analysis method by means of integral equations and boundary elements. Meccanica 14 (1979) 210–218.
40. O. Steinbach: A robust boundary element method for nearly incompressible linear elasticity. Numer. Math. 95 (2003) 553–562.
41. O. Steinbach: Artificial multilevel boundary element preconditioners. Proc. Appl. Math. Mech. 3 (2003) 539–542.
42. O. Steinbach: Numerische Näherungsverfahren für elliptische Randwertprobleme. Finite Elemente und Randelemente. B.G. Teubner, Stuttgart, Leipzig, Wiesbaden, 2003.
43. O. Steinbach, W. L. Wendland: The construction of some efficient preconditioners in the boundary element method. Adv. Comput. Math. 9 (1998) 191–216.
44. J. Tausch: The variable order fast multipole method for boundary integral equations of the second kind. Computing 72 (2004) 267–291.
45. C. A. White, M. Head-Gordon: Derivation and efficient implementation of the fast multipole method. J. Chem. Phys. 101 (1994) 6593–6605.
46. K. Yoshida, N. Nishimura, S. Kobayashi: Application of fast multipole Galerkin boundary integral equation method to elastostatic crack problems in 3D. Internat. J. Numer. Methods Engrg. 50 (2001) 525–547.