ELSEVIER

Engineering Analysis with Boundary Elements 27 (2003) 423–437

ENGINEERING ANALYSIS *with* BOUNDARY ELEMENTS

www.elsevier.com/locate/enganabound

# Fully discrete wavelet Galerkin schemes

Helmut Harbrecht*, Michael Konik, Reinhold Schneider

*Fakultät für Mathematik, Technische Universität Chemnitz, D-09107 Chemnitz, Germany*

Received 1 February 2002; revised 28 March 2002; accepted 22 April 2002

**Abstract**

The present paper is intended to give a survey of the developments of the wavelet Galerkin boundary element method. Using appropriate wavelet bases for the discretization of boundary integral operators yields numerically sparse system matrices. These system matrices can be compressed to $\mathcal{O}(N_j)$ nonzero matrix entries without loss of accuracy of the underlying Galerkin scheme. Herein, $\mathcal{O}(N_j)$ denotes the number of unknowns. As we show in the present paper, the assembly of the compressed system matrix can be performed within optimal complexity. By numerical experiments we provide examples which corroborate the theory.

*Keywords:* Galerkin boundary element method; Sparse system matrix; Hierarchical matrix approach; wavelets

## 1. Introduction

Widely used approaches for the solution of elliptic boundary value problems in the three-dimensional Euclidean spaces are, e.g. the discretization by finite differences or finite elements. In particular, for exterior boundary value problems these methods encounter serious problems concerning the discretization of an infinite domain. An alternative approach is the boundary element method, which transfers the problem to the boundary of the given domain. This reduces the dimension of the problem, and furthermore, limits the discretization to the surface of the domain. Unfortunately, the resulting system matrix is densely populated and possibly ill-conditioned. This makes the computation very costly in both respects, the computation time and computer memory requirements. The dense system matrix leads to algorithms which computational costs are at least the square of the number of unknowns.

In recent years several ideas for the efficient approximation of the discrete system have been developed. All these methods have in common a fast matrix–vector multiplication combined with the use of iterative linear solvers. Most prominent examples of these methods are fast multipole [21], panel clustering [25], wavelet Galerkin methods [2,15], mosaic skeleton approaches [20] and the hierarchical matrix approach [23,24]. Such fast discretization methods end up with linear or almost linear complexity with respect to the number of unknowns.

In this paper we present a fully discrete wavelet Galerkin approach. For this approach it is possible to develop an algorithm with linear complexity (without any logarithmic factor) by preserving the optimal order of convergence of the underlying Galerkin scheme [29,44]. In the wavelet Galerkin method we use a wavelet basis for the representation of the Galerkin scheme. The arising matrix can be approximated by a sparse matrix. The sparsity pattern is chosen carefully by a level dependent compression strategy such that the optimal order of convergence of the Galerkin scheme is not violated [12,40]. Our numerical experiences confirm that the accuracy of the Galerkin scheme is never deteriorated if we use the present compression strategy. Sometimes, due to round-off errors the compressed scheme behaves even slightly better. In our numerical calculations we observe rates of compression up to a factor of 1000 for linear systems with about 200,000 unknowns [26]. Our present examples at the end of the paper demonstrate a similar efficiency even for nontrivial geometries.

Our fully discrete scheme is based on numerical quadrature. It has been proved that for this method the number of quadrature points grows also only linearly with the number of unknowns. But the computation of the coefficients of the sparse matrix approximation involves integrals in higher dimensions and is still time-consuming. Depending on the relative distance between the supports of

* Corresponding author.

0955-7997/03/$ - see front matter © 2003 Elsevier Science Ltd. All rights reserved.
doi:10.1016/S0955-7997(02)00153-4

the wavelets we have three types of integrals, namely singular integrals, nearly singular integrals and farfield integrals. We have to deal with singular integrals for which we propose special quadrature techniques on cubes based on the work in Refs. [19,38,39]. The efficient approximation of the so-called nearly singular integrals, where the domains of integration are very close together, are treated like in Refs. [40,41]. By the proposed method the solution is computed within asymptotically linear complexity without compromising the accuracy of the Galerkin scheme.

Nevertheless we are confronted with some difficulties which have to be considered in order to get an efficient realization on a computer. In our present implementations the discretization of the surface of a three-dimensional domain is represented by piecewise parametric mappings of a two-dimensional reference domain—a well studied tool in Computer Aided Geometric Design. For extremely complex geometries the number of unknowns are not yet satisfactory and thus the treatment of the geometry in connection with wavelet methods is still in progress. The computation of suitable multiscale bases on surface patches is addressed in several publications [17,26,32,40].

Wavelet matrix compression has been introduced by Beylkin et al. [2]. Compression strategies which retain the order of convergence have been investigated in Refs. [15,16,33,34]. The remaining logarithmic term in the complexity has been removed by the so-called second compression [40]. Appropriate wavelet bases for boundary integral equations on surfaces have been developed in Refs. [5,17,18,32]. The efficient computation of the compressed system matrices has been addressed in Refs. [32,35,40]. This development has been the guideline for the implementation of a fully discrete wavelet Galerkin scheme. A first successful realization has been presented in Ref. [31] using multi-wavelets for integral equations of second kind. The present paper is intended to give a survey of wavelet Galerkin methods and their recent developments [26,30]. Several ingredients are important to achieve linear complexity for boundary integral operators arising from indirect or direct formulations, namely biorthogonal wavelet bases [17], fully discrete wavelet Galerkin methods [31,34,40], the second compression [40], and the wavelet preconditioning [13]. In contrast to previous attempts our realization achieves linear complexity also for integral operators of nonzero order. For sake of brevity we refer to Refs. [26,30] for the technical details. We like to mention recent developments of adaptive wavelet methods in Ref. [6]. These methods are based on the concept of the best $N$-term approximation and offer further perspectives for future developments.

## 2. Setting up the problem

For the numerical approximation of a boundary integral equation we need a discretization method which ends up with a sufficiently accurate finite-dimensional approximation of the given operator. At first we consider a general setting for the boundary element method. Next, a short description of the representation of the geometry on a computer is given. Then, we discuss the properties for the class of kernel functions under consideration.

The further outline is as follows. In Section 3 we give a brief introduction to multiscale bases. With the help of such a basis we can describe the multiscale Galerkin discretization in Section 4. A multiscale discretization in the framework of a Galerkin method allows to exploit the advantages of wavelet bases. The compact support of wavelets offers the possibility to focus on local phenomena of the discretized operator and their good approximation properties allow to reduce the number of unknowns drastically. We concentrate on wavelet bases based on cardinal B-spline wavelets. With such a basis at hand we are in the position to describe the classical multiscale Galerkin method. We recall the main advantages of the wavelet basis which result in the matrix compression algorithm [40].

In the last years wavelet approaches have been developed to reduce the number of unknowns drastically in various applications. The numerical treatment of boundary integral equations in connection with wavelets benefits from these compression techniques. A multiscale ansatz is based on a discrete approximation of the operator on a relatively coarse approximation level and adding the details of subsequent levels. In Section 5 we present the compression algorithm which reduces the relevant matrix coefficients to an asymptotically linear number [40]. In Section 6 we show that also the number of function evaluations to compute those matrix entries grows only linearly with respect to the number of unknowns [12].

Section 7 is dedicated to the preconditioning of the system matrices arising from boundary integral operators with nonzero order. In the wavelet basis we have the possibility to precondition the system matrix by diagonal scaling [13]. Finally, in Section 8 we present numerical computations which confirm the theory quite well.

### 2.1. Boundary integral equations

We consider a boundary integral equation on the closed boundary surface $\Gamma$ of an $(s+1)$-dimensional domain $\Omega \subset \mathbb{R}^{s+1}$

$$Au(x) = \int_\Gamma k(x,y)u(y)\mathrm{d}\Gamma_y = f(x), \qquad x \in \Gamma. \tag{1}$$

Especially we are interested in the case $s = 2$. For the present purpose, we assume that the boundary $\Gamma$ is a two-dimensional surface in $\mathbb{R}^3$, which is represented by piecewise parametric mappings (Section 2.2). The number of different mappings, which is the number of surface patches, will be denoted by $N_\pi$. In this paper we will also examine to which extent the present multiscale approach can be applied efficiently. The surface representation is in contrast to the usual approximation of the surface by panels. It has the advantage that the rate of convergence is not limited by this approximation. Notice that technical surfaces generated by CAD tools are represented in this form. Of course, this fact makes the use of numerical integration indispensable for the computation of the system matrices.

**Example 2.1 (Single layer potential operator of order $r = -1$).** We consider the boundary integral equation of first kind

$$Vu(x) = \int_\Gamma k(x,y)u(y)\mathrm{d}\Gamma_y = f(x), \qquad x \in \Gamma.$$

The right hand side $f$ resulting from the direct approach [9,10] is given by $f = (\frac{1}{2}I + K)g$ where $K$ is the double layer potential operator

$$Ku(x) = \int_\Gamma \frac{\partial}{\partial n_y} k(x,y)u(y)\mathrm{d}\Gamma_y, \qquad x \in \Gamma. \tag{2}$$

- In three dimensions we have

$$k(x,y) = \frac{1}{4\pi}\frac{1}{|x-y|}$$

- and in two dimensions the kernel function is defined by

$$k(x,y) = -\frac{1}{2\pi}\log|x-y|.$$

It is known that $V$ maps the Sobolev space $H^{-1/2}(\Gamma)$ into $H^{1/2}(\Gamma)$ and leads to a symmetric elliptic variational problem. The operator $V^{-1}(\frac{1}{2}I + K) : H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$ transfers the Dirichlet data of harmonic functions to the Neumann data (i.e. $\Delta u(x) = 0$ on $\Omega$).

The properties of the class of kernel functions $k(x,y)$ which are under consideration will be outlined in Section 2.3. Further examples of boundary integral equations are considered in the numerical results (Section 8).

## 2.2. Parametric representation of geometry

The construction of a wavelet basis $\Psi$ on a manifold $\Gamma$ depends essentially on the way how the boundary $\Gamma$ is represented. The geometry is supposed to be exactly defined by the union of parametric surface patches. This setting is frequently used and well understood in Computer Aided Geometric Design, cf. Refs. [28,37].

We describe a surface patch $\pi_i$ as the image of a reference domain (e.g. $\square := [0,1]^s$) by a parametric mapping $\kappa_i$. The boundary $\Gamma$ is represented by the union of sufficiently many patches

$$\Gamma = \bigcup_{i=1}^{N_\pi} \pi_i, \qquad \pi_i = \kappa_i(\square), \;\; i = 1,\dots,N_\pi.$$

Such surface patches are not allowed to share any interior points, i.e. they either have a common vertex, a common edge or an empty intersection. An example of a parametrization of a drilled out cube can be found in Fig. 1.

The parametric mappings $\kappa_i : \mathbb{R} \to \mathbb{R}^{s+1}$ are supposed to be smooth functions. In many practical applications the parametrizations are of sufficiently high componentwise smoothness, e.g. piecewise rational or polynomial functions.

## 2.3. Kernel functions and their properties

The class of kernel functions under consideration are functions in two variables which are smooth apart from the diagonal and may have a singularity on the diagonal. Such kernel functions arise by applying a boundary integral formulation to a second order elliptic problem, for example. In general, they decay like a negative power of the distance of the arguments, which depends on the spatial dimension $s$ and the order $r$ of the operator.

We denote by $|\kappa_x|$ and $|\kappa_y|$ the Jacobian determinants of the parametric map. Moreover, $\alpha$ and $\beta$ encode multi-indices of dimensions and $|\alpha| := \alpha_1 + \cdots + \alpha_s$. Furthermore, $\hat{x}$ and $\hat{y}$ are points on the surface, i.e. $\hat{x} := \kappa(x)$.

**Definition 2.2.** A kernel $k(\hat{x},\hat{y})$ is called analytically standard of order $r$, if the partial derivatives of

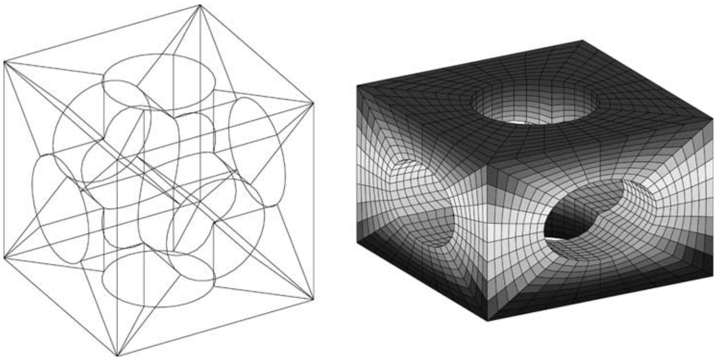

Fig. 1. The domain decomposition of a drilled out cube into 48 patches (left) and the corresponding mesh after three subdivision steps (right).

the transported kernel function

$$\mathscr{K}(x,y) := k(\kappa(x),\kappa(y))|\kappa_x||\kappa_y| \tag{3}$$

are uniformly bounded by

$$|\partial_x^\alpha \partial_y^\beta \mathscr{K}(x,y)| \le C\frac{(|\alpha|+|\beta|)!}{(q\ \mathrm{dist}(\hat{x},\hat{y}))^{s+r+|\alpha|+|\beta|}}, \tag{4}$$

$$\hat{x} := \kappa(x), \qquad \hat{y} := \kappa(y),$$

with some $q > 0$.

We emphasize that this definition requires patchwise smoothness but *not* global smoothness of the geometry. The surface itself needs only to be Lipschitz. Generally under this assumption, the kernel of a boundary integral operator of order $r$ is analytically standard of order $r$. Hence, we may assume this property in the sequel.

## 3. Multiscale bases

In this section we will recall some basic framework of wavelet analysis. Clearly, of main interest are cardinal B-spline wavelets because they offer some additional features which are very important in the analysis of a fully discrete wavelet Galerkin scheme for boundary integral equations. For cardinal B-spline wavelets we can find a biorthogonal system such that the wavelets on the dual side have higher polynomial power of approximation as on the primal side [7,14,17]. This flexibility is widely used and necessary to get a wavelet Galerkin algorithm with linear complexity by retaining the optimal order of convergence of the Galerkin scheme.

The transform from the single-scale into the multiscale basis can be performed very fast for all wavelets due to their local supports. In the case of cardinal B-spline wavelets it is possible to find dual wavelets having also compact supports. This implies a fast back transform from the multiscale into the single-scale basis.

The wavelets on the surface are formed, for example, by taking tensor products of one-dimensional interval wavelets [14,17] which are lifted via the parametrization onto the surface patches. Improved definitions can be found in Ref. [26]. In Ref. [17], the basis functions near the boundary of a patch are modified to get global smoothness. Similar constructions based on domain decomposition can be found in Refs. [3,8].

### *3.1. Scaling functions*

For the Galerkin scheme we replace the original equation (1) by a finite-dimensional approximation. Therefore, we need finite-dimensional function spaces $S_j$ in $L_2(\Gamma)$. Those function spaces $S_j$ are generated by the so-called *single-scale basis* on a certain arbitrary but fixed level $j$

$$\mathbf{\Phi}^j = \{\varphi_\lambda : \lambda \in \Delta_j\}, \qquad \Delta_j := \{\lambda : |\lambda| = j\}.$$

The multi-index $\lambda := (j,k)$ contains the information which is necessary to address a basis function on the surface in a unique way. With $|\lambda| := j$ we encode the level of the function while $k$ denotes its location. We mention that the functions in $\mathbf{\Phi}^j$ can be addressed by the indices $\lambda$ in several ways, e.g. as a multi-index or by associated points.

Of course, for the Galerkin scheme, the basis functions are required to provide a certain power of approximation $d$, that is

$$\inf_{f_j \in S_j} \|f - f_j\|_0 \le C2^{-jd}\|f\|_{W^{d,\infty}(\Gamma)}. \tag{5}$$

We assume further that the basis functions $\varphi_\lambda \in \mathbf{\Phi}^j$ have compact supports and that the size of the supports behaves like $2^{-j}$, i.e.

$$\mathrm{diam\ supp}\ \varphi_\lambda \sim 2^{-j} \qquad \text{for } j \to \infty.$$

The locality of the basis functions is an important property, which allows to focus on the local behaviour of the underlying operator. Furthermore, it is convenient to consider normalized basis functions $\|\varphi_\lambda\|_0 \sim 1$. Here $a \sim b$ means that $a$ is bounded from above and below by $c_1 b \le a \le c_2 b$, with some constants $c_1$, $c_2$ which are independent of $a$ and $b$.

In the application of wavelet Galerkin methods for boundary integral equations it is necessary to have access to a suitable *biorthogonal* wavelet basis. The less restrictive biorthogonal construction in comparison to the orthogonal one offers some flexibility in choosing the number of vanishing moments. Furthermore, we can retain this power of approximation at the boundary of a patch or to construct a globally continuous wavelet basis on the surface. The biorthogonal basis is generated by a second system $\{\tilde{\varphi}_{\lambda'} : \lambda' \in \Delta_j\}$ for the given basis $\{\varphi_\lambda : \lambda \in \Delta_j\}$ satisfying

$$(\varphi_\lambda, \tilde{\varphi}_{\lambda'}) = \delta_{\lambda,\lambda'}, \qquad \text{for } \lambda, \lambda' \in \Delta_j. \tag{6}$$

The spaces $\tilde{S}_j$ spanned by the biorthogonal basis functions $\tilde{\varphi}_{\lambda'}$ are finite-dimensional spaces with $\dim S_j = \dim \tilde{S}_j$.

With the biorthogonal basis at hand we are in the position to realize finite-dimensional approximations of functions in $L_2(\Gamma)$ in terms of projectors of the form

$$Q_j f := \sum_{\lambda \in \Delta_j} (f, \tilde{\varphi}_\lambda)\varphi_\lambda. \tag{7}$$

The basis functions $\varphi_\lambda$ are supposed to form a stable basis in the sense that the following norm equivalence holds:

$$\left\|\sum_{\lambda \in \Delta_j} u_\lambda \varphi_\lambda\right\|_0^2 \sim \sum_{\lambda \in \Delta_j} |u_\lambda|^2.$$

Note that up to now we have a single-scale basis in $S_j$ for an arbitrary but fixed level.

### 3.2. Wavelets

Multiscale concepts are based on a sequence of nested discretization spaces $S_l$ such that

$$S_0 \subset \cdots \subset S_l \subset S_{l+1} \subset \cdots \subset L_2(\Gamma)$$

and, moreover, we assume that the dual spaces $\tilde{S}_l$ are also nested, i.e.

$$\tilde{S}_0 \subset \cdots \subset \tilde{S}_l \subset \tilde{S}_{l+1} \subset \cdots \subset L_2(\Gamma). \tag{8}$$

Using the projectors $Q_j$ from Eq. (7) we obtain an approximation in $S_j := Q_j L_2(\Gamma)$ for each function $f \in L_2(\Gamma)$. We introduce the linear spaces

$$W_{j-1} := (Q_j - Q_{j-1}) L_2(\Gamma)$$

and get

$$S_j = Q_j L_2(\Gamma) = (Q_j - Q_{j-1}) L_2(\Gamma) + Q_{j-1} L_2(\Gamma)$$
$$= W_{j-1} + S_{j-1}.$$

Here we use the summation sign for the direct sum of vector spaces, i.e. $W_{j-1} \cap S_{j-1} = \{0\}$. Then the *multiscale decomposition* of $S_j$ is given by

$$S_j = \sum_{l=-1}^{j-1} W_l$$

with $W_{-1} := S_0$. A function $f_j \in S_j$ can be represented by the telescopic sum

$$f_j = \sum_{l=0}^{j} (Q_l - Q_{l-1}) f \qquad \text{with } Q_{-1} := 0. \tag{9}$$

The assumption (8) is equivalent to the fact that $Q_l Q_j = Q_l$ for $l \le j$ and that $(Q_l - Q_{l-1})$ is also a projector [40]. The second property ensures that we can find basis functions such that

$$(Q_l - Q_{l-1}) f = \sum_{\lambda \in \nabla_{l-1}} (f, \tilde{\psi}_\lambda) \psi_\lambda$$

with the index set $\nabla_l := \Delta_{l+1} \backslash \Delta_l$ and $\nabla_{-1} := \Delta_0$. On each level we have a basis $\mathbf{\Psi}^l = \{\psi_\lambda : \lambda \in \nabla_l\}$ for the complementary spaces $W_l$ of $S_l$ in $S_{l+1}$. We assume that the function $\psi_\lambda$ has also compact supports with respect to the associated level

$$\operatorname{diam} \operatorname{supp} \psi_\lambda \sim 2^{-|\lambda|}.$$

Furthermore, there exists a set of basis functions $\tilde{\psi}_{\lambda'}$, $\lambda' \in \nabla_l$, $l \ge -1$, spanning the spaces $\tilde{S}_l$ which are biorthogonal to the basis functions $\psi_\lambda$, i.e.

$$(\tilde{\psi}_{\lambda'}, \psi_\lambda) = \delta_{\lambda',\lambda}.$$

We obtain the *multiscale basis* for $S_j$ by

$$\mathbf{\Psi} := \bigcup_{l=-1}^{j-1} \mathbf{\Psi}^l$$

using the notation $\mathbf{\Psi}^{-1} := \mathbf{\Phi}^0$. The basis functions in $\boldsymbol{\psi}$ are called *wavelets*.

Now we have two bases for the space $S_j$, the single-scale basis and the wavelet basis. These allows us to express the projection of the function $f \in L_2(\Gamma)$ onto $S_j$ by

$$f_j = \sum_{\lambda \in \Delta_j} f_\lambda \varphi_\lambda = \sum_{l=-1}^{j-1} \sum_{\lambda \in \nabla_l} d_\lambda \psi_\lambda.$$

To ensure that the wavelet transform from the multiscale to the single-scale basis and its inverse are linear in complexity we require that the primal as well as the dual counterparts of wavelet and scaling functions have also compact support, i.e.

$$\operatorname{diam} \operatorname{supp} \tilde{\varphi}_{\lambda'} \sim 2^{-|\lambda'|} \qquad \text{and} \qquad \operatorname{diam} \operatorname{supp} \tilde{\psi}_{\lambda'} \sim 2^{-|\lambda'|}.$$

Moreover, for the matrix compression the wavelets are requested to have a certain number of vanishing moments $d^*$ which is also called *cancellation property*

$$|(f, \psi_\lambda)| \le C 2^{-l(d^* + (s/2))} |f|_{W^{d^*,\infty}(\Gamma)}. \tag{10}$$

Herein, $|\cdot|_{W^{d^*,\infty}(\Gamma)}$ denotes the usual $L^\infty$-Sobolev semi-norm. Note that $d^*$ denotes the power of approximation of the dual wavelet.

Wavelet constructions based on cardinal B-splines and their duals fulfill the requirements mentioned above. Other approaches in the direction of wavelet Galerkin methods work with multi-wavelets [32]. Globally continuous basis functions with such properties are constructed in Refs. [14, 17], see also Fig. 2. The construction is based on a stable completion [4,40].

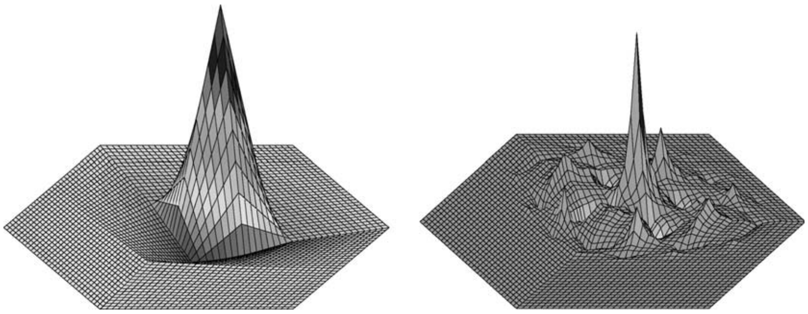

Fig. 2. A globally continuous piecewise linear wavelet with two vanishing moments (left) and its corresponding dual (right) near a degenerated vertex.

$$\begin{tikzcd}
 & \boldsymbol{d}^{j-1} \arrow[dl, "\boldsymbol{M}_{j,1}"] & \boldsymbol{d}^{j-2} \arrow[dl, "\boldsymbol{M}_{j-1,1}"] & {} \arrow[dl] & \dots & \boldsymbol{d}^{1} \arrow[dl, "\boldsymbol{M}_{1,1}"] & \boldsymbol{d}^{0} \arrow[dl, "\boldsymbol{M}_{0,1}"] \\
\boldsymbol{c}^{j} & \boldsymbol{c}^{j-1} \arrow[l, "\boldsymbol{M}_{j,0}"'] & \boldsymbol{c}^{j-2} \arrow[l, "\boldsymbol{M}_{j-1,0}"'] & {} \arrow[l] & \dots & \boldsymbol{c}^{1} \arrow[l, "\boldsymbol{M}_{1,0}"'] & \boldsymbol{c}^{0} \arrow[l, "\boldsymbol{M}_{0,0}"']
\end{tikzcd}$$

Fig. 3. Reconstruction.

### 3.3. Basis transforms

The nestedness of spaces $S_l \subset S_{l+1}$ implies that the basis functions in $S_l$ can be expressed by a linear combination of basis functions in $S_{l+1}$, the so-called *two-scale* or *refinement equation*

$$\varphi_\lambda = \sum_{\mu \in \Delta_{l+1}} a_{\mu,\lambda} \varphi_\mu, \qquad \lambda \in \Delta_l. \tag{11}$$

Eq. (11) can be written in matrix notation as

$$\mathbf{\Phi}_l^{\mathrm{T}} = \mathbf{\Phi}_{l+1}^{\mathrm{T}} \mathbf{M}_{l,0} \qquad \text{with } \mathbf{M}_{l,0} := (a_{\mu,\lambda})_{\mu \in \Delta_{l+1}, \lambda \in \Delta_l}. \tag{12}$$

Analogously and because of $W_l \subset S_{l+1}$ there also exists a functional equation for representing the wavelets in terms of scaling functions on the next higher level

$$\psi_\lambda = \sum_{\mu \in \Delta_{l+1}} b_{\mu,\lambda} \varphi_\mu, \qquad \lambda \in \Delta_l \tag{13}$$

or

$$\mathbf{\Psi}_l^{\mathrm{T}} = \mathbf{\Phi}_{l+1}^{\mathrm{T}} \mathbf{M}_{l,1}, \qquad \mathbf{M}_{l,1} := (b_{\mu,\lambda})_{\mu \in \Delta_{l+1}, \lambda \in \nabla_l}. \tag{14}$$

On the basis of the nestedness of the dual spaces similar functional equations hold true. We have

$$\tilde{\varphi}_\lambda = \sum_{\mu \in \Delta_{l+1}} \tilde{a}_{\mu,\lambda} \tilde{\varphi}_\mu, \ \ \lambda \in \Delta_l \qquad \text{or } \tilde{\mathbf{\Phi}}_l = \mathbf{G}_{l,0}^{\mathrm{T}} \tilde{\mathbf{\Phi}}_{l+1}$$

as well as

$$\tilde{\psi}_\lambda = \sum_{\mu \in \nabla_{l+1}} \tilde{b}_{\mu,\lambda} \tilde{\varphi}_\mu, \ \ \lambda \in \nabla_l, \qquad \text{or } \tilde{\mathbf{\Psi}}_l = \mathbf{G}_{l,1}^{\mathrm{T}} \tilde{\mathbf{\Psi}}_{l+1}.$$

We get transform matrices $\mathbf{G}_l = (\mathbf{G}_{l,0}, \mathbf{G}_{l,1})$ and $\mathbf{M}_l = (\mathbf{M}_{l,0}, \mathbf{M}_{l,1})$ to perform one step of the multiscale transform by

$$\mathbf{\Phi}_{l+1} = \mathbf{G}_l \begin{pmatrix} \mathbf{\Phi}_l \\ \mathbf{\Psi}_l \end{pmatrix}, \tag{15}$$

respectively,

$$\begin{pmatrix} \mathbf{\Phi}_l \\ \mathbf{\Psi}_l \end{pmatrix}^{\mathrm{T}} = \mathbf{\Phi}_{l+1}^{\mathrm{T}} \mathbf{M}_l.$$

The biorthogonality then implies that

$$\mathbf{G}_{l,e}^{\mathrm{T}} \mathbf{M}_{l,e'} = \delta_{e,e'} \mathbf{I} \qquad \text{with } e, e' \in \{0,1\}.$$

One step of the multiscale decomposition can be expressed by Eqs. (12) and (14) as

$$\mathbf{\Phi}_l^{\mathrm{T}} \mathbf{c}_l + \mathbf{\Psi}_l^{\mathrm{T}} \mathbf{d}_l = \mathbf{\Phi}_{l+1}^{\mathrm{T}} (\mathbf{M}_{l,0} \mathbf{c}_l + \mathbf{M}_{l,1} \mathbf{d}_l).$$

The whole transform from a single-scale basis to a multiscale basis is expressed by matrix multiplications as follows. If we define the matrix

$$\mathbf{T}_{j,l} := \begin{pmatrix} \mathbf{M}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{j-l} \end{pmatrix}$$

for $0 \le l < j$ then the multiscale transform is given by

$$\mathbf{T} = \mathbf{T}_{j,0} \cdot \cdots \cdot \mathbf{T}_{j,j-1}.$$

This procedure realizes the reconstruction scheme shown in Fig. 3.

In view of Eq. (15) the inverse transform reads as

$$\mathbf{\Phi}_{l+1}^{\mathrm{T}} \mathbf{c}_{l+1} = \mathbf{\Phi}_l^{\mathrm{T}} (\mathbf{G}_{l,0} \mathbf{c}_{l+1}) + \mathbf{\Psi}_l^{\mathrm{T}} (\mathbf{G}_{l,1} \mathbf{c}_{l+1}) = \mathbf{\Phi}_l^{\mathrm{T}} \mathbf{c}_l + \mathbf{\Psi}_l^{\mathrm{T}} \mathbf{d}_l$$

and corresponds to the multiscale decomposition as shown in Fig. 4.

## 4. Multiscale Galerkin method

In this section we describe the finite-dimensional approximation process of the boundary integral equation (1) using a multiscale Galerkin method. We have already introduced the spaces $\{S_l\}_{l \in \mathbb{N}}$ and $\{\tilde{S}_l\}_{l \in \mathbb{N}}$. In the sequel we will assume these families of finite-dimensional subspaces to be generated by a cardinal B-Spline function for arbitrary but fixed $d$, $d^*$ ($d + d^*$ even). This may include adaption near the interfaces of the patches.

A given function can be represented in a multiscale basis by a telescopic sum of projections (9) onto the finite-dimensional spaces $S_l$. To discretize the original operator equation

$$Au = f,$$

$$\begin{tikzcd}
 & \boldsymbol{d}^{j-1} & \boldsymbol{d}^{j-2} & {} & \dots & \boldsymbol{d}^{1} & \boldsymbol{d}^{0} \\
\boldsymbol{c}^{j} \arrow[r, "\boldsymbol{G}_{j,0}"] \arrow[ur, "\boldsymbol{G}_{j,1}"'] & \boldsymbol{c}^{j-1} \arrow[r, "\boldsymbol{G}_{j-1,0}"] \arrow[ur, "\boldsymbol{G}_{j-1,1}"'] & \boldsymbol{c}^{j-2} \arrow[r] \arrow[ur] & {} & \dots \arrow[r, "\boldsymbol{G}_{1,0}"] \arrow[ur, "\boldsymbol{G}_{1,1}"'] & \boldsymbol{c}^{1} \arrow[r, "\boldsymbol{G}_{0,0}"] \arrow[ur, "\boldsymbol{G}_{0,1}"'] & \boldsymbol{c}^{0}
\end{tikzcd}$$

Fig. 4. Decomposition.

we replace the solution $u$ by a function in $S_j$ and apply to both sides of the equation the adjoint projector $Q_j^*$

$$A_j u_j := Q_j^* A Q_j u_j = Q_j^* f. \tag{16}$$

The approximate solution $u_j$ converges to the original solution $u$ for $j \to \infty$ if and only if the discrete operator $A_j$ is consistent to the original operator $A$ and if the discrete operator is stable in the sense that we have a uniform a priori estimate $\|A_j u_j\| \geq C\|u_j\|$. It is proven in Ref. [40] that both are valid in the present setting.

The Eq. (16) reads in the multiscale basis as

$$\sum_{l'=0}^{j} \sum_{l=0}^{j} (Q_{l'}^* - Q_{l'-1}^*) A (Q_l - Q_{l-1}) u_j = \sum_{l'=0}^{j} (Q_{l'}^* - Q_{l'-1}^*) f$$

respectively, as

$$\sum_{l'=-1}^{j-1} \sum_{l=-1}^{j-1} \sum_{\lambda' \in \nabla_{l'}} \sum_{\lambda \in \nabla_l} (A\psi_\lambda, \psi_{\lambda'})(u_j, \tilde{\psi}_\lambda)\tilde{\psi}_{\lambda'}$$

$$= \sum_{l'=-1}^{j-1} \sum_{\lambda' \in \nabla_{l'}} (f, \psi_{\lambda'})\tilde{\psi}_{\lambda'}.$$

This representation results in the linear system of equations

$$\mathbf{A}_j \mathbf{d}_j = \mathbf{b}_j.$$

It is equivalent to find $u_j \in S_j$ such that

$$(Au_j, v_j) = (f, v_j) \qquad \forall v_j \in S_j.$$

The multiscale matrix $\mathbf{A}_j$ contains the entries

$$\mathbf{A}_j = (a_{\lambda,\lambda'}) = (A\psi_{\lambda'}, \psi_\lambda),$$

$$\lambda \in \nabla_l, \quad \lambda' \in \nabla_{l'}, \quad -1 \leq l, l' < j$$

and the vectors $\mathbf{d}_j$ and $\mathbf{b}_j$ have the following form

$$\mathbf{d}_j = (d_\lambda) = (u, \tilde{\psi}_\lambda), \qquad \lambda \in \nabla_l, \quad -1 \leq l < j,$$

$$\mathbf{b}_j = (b_{\lambda'}) = (f, \psi_{\lambda'}), \qquad \lambda' \in \nabla_{l'}, \quad -1 \leq l' < j,$$

The system matrix $\mathbf{A}_j$ in the multiscale basis is still dense. Nevertheless, it is shown in Refs. [15,40] that most coefficients in the multiscale matrix are close to zero and negligible without any significant loss of accuracy. We call such a matrix *numerically sparse*. The process of approximating the system matrix by a sparse matrix is called *matrix compression*.

## 5. Matrix compression

It is shown in Ref. [40] that the matrix entries fulfill the estimate

$$|a_{\lambda,\lambda'}| \leq C \frac{2^{-(l+l')(d^*+s/2)}}{\operatorname{dist}(\Omega_\lambda, \Omega_{\lambda'})^{s+r+2d^*}} \tag{17}$$

with the order $r$ from the integral operator in Eq. (4). We denote by $\Omega_\lambda := \operatorname{supp} \psi_\lambda$ the support of the wavelet function on the surface.

This above estimate is based on the Taylor expansion of the kernel function and the use of the cancellation property (10) of the wavelets. The arising derivatives in Taylor's expansion can be estimated by the decay property of the kernel function (4) [40].

### 5.1. First compression

Eq. (17) implies that the coefficients in the matrix block $\mathbf{A}_{l,l'}$ decline when the distance of the supports increases. The matrix block $\mathbf{A}_{l,l'}$ contains all those coefficients for which $|\lambda| = l$ and $|\lambda'| = l'$. Based on the observation (17) it is possible to perform a compression step, the so-called *first compression*, by neglecting all coefficients where the distance of the supports of the associated wavelets is larger than a level dependent truncation parameter. The compression algorithm can be improved by the *second compression* where we additionally neglect some coefficients for which the difference of the levels is sufficiently large. After the matrix compression there remain only $\mathcal{O}(N_j)$ matrix entries. They are sufficient to compute an appropriately accurate solution retaining the optimal order of convergence of the Galerkin scheme. Here $N_j \sim 2^{js}$ denotes the number of unknowns for a discretization with the maximal level $j$. The level dependent bandwidth is required to ensure this optimal order of convergence. We intend to compute only these $\mathcal{O}(N_j)$ matrix entries including the relevant matrix coefficients reflecting the essential information of the operator.

We will show in Section 6 that under the above assumptions we can compute the compressed matrix with at most $\mathcal{O}(N_j)$ function evaluations [12,26,30]. The outline is as follows. First, we introduce the first compression which reduces the number of nonzero coefficients in the system matrix to at most $\mathcal{O}(N_j \log N_j)$ (Lemma 5.1). Then, Lemma 5.3 confirms that by the second compression the number of matrix entries can be reduced even to $\mathcal{O}(N_j)$ nonzero coefficients. The Section 5.3 explains how to set up the compression pattern retaining this optimal complexity.

The analysis in Ref. [40] shows that only coefficients where the supports of the wavelets are close to each other or are overlapping are required. Consequently, an entry of the system matrix has to be calculated if

$$\operatorname{dist}(\Omega_\lambda, \Omega_{\lambda'}) < \mathcal{B}_{l,l'} \tag{18}$$

where

$$\mathcal{B}_{l,l'} = a \max\{2^{-\min\{l,l'\}}, 2^{[j(2d'-r)-(l+l')(d^*+d')]/(2d^*+r)}\},$$

$$d < d' < d^* + r, \quad a > 1$$

is a level dependent bandwidth. The parameter $a$ has to be chosen appropriately. Then we obtain a blockmatrix of bandmatrices (Fig. 5) in the two-dimensional case or

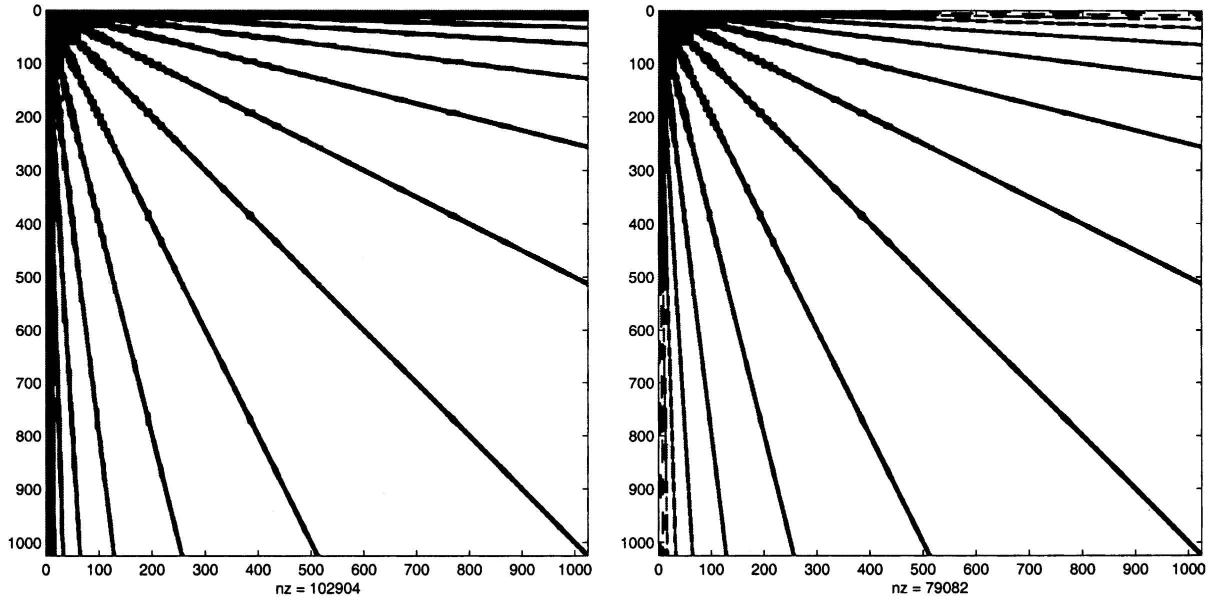

Fig. 5. The sparsity pattern of the system matrix after the first compression (left) and after the first and second compression (right).

a blockmatrix of sparse matrices with a number of bands in a more complicated setting (3D case). In the system matrix remain $\mathcal{O}(N_j \log N_j)$ coefficients [40]. The result is given in the following lemma.

**Lemma 5.1.** *Suppose that* $d < d^* + r$ *and let* $d'$ *be an appropriate chosen parameter such that* $d < d' < d^* + r$. *Let* $\mathbf{A}_j^\varepsilon$ *be the resulting matrix after the first compression, which means neglecting all matrix entries with*

$$\operatorname{dist}(\Omega_\lambda, \Omega_{\lambda'}) \geq \mathscr{B}_{l,l'}$$

*and*

$$\mathscr{B}_{l,l'} = a \max\{2^{-\min\{l,l'\}}, 2^{[j(2d'-r)-(l+l')(d^*+d')]/(2d^*+r)}\}.$$

*Then the matrix* $\mathbf{A}_j^\varepsilon$ *has at most* $\mathcal{O}(N_j \log N_j)$ *nonzero coefficients.*

Note that only the entries with $\Omega_\lambda \cap \Omega_{\lambda'} = \varnothing$ are affected by the first compression (Fig. 6).

**Remark 5.2.** In theory the parameter $a$ has to be chosen sufficiently large in order to ensure the stability of the compressed linear system. For numerical calculations, the parameter has to be adjusted on low levels such that the compression does not violate the accuracy. However, we made the experience that the simple choice $a = 1$ is sufficient when choosing the length-scale such that the largest patch has the diameter $\sqrt{2}$ [26]. We used this choice for the computations in Section 8. We mention that the parameter $a$ plays a similar role as the ratio of the diameters of clusters and their distance in the panel clustering method. So far, we have not considered extreme nonuniform domain decompositions and very anisotropic geometries.

### *5.2. Second compression*

A second compression in this section affects all coefficients which come out of the first compression step. Especially it reduces the number of coefficients in those blocks where $\Omega_\lambda \subset \Omega_{\lambda'}$ and $|\lambda| \gg |\lambda'|$ is valid.

In the case of $d < d' < d^* + r$, we apply a second compression which allows to neglect even more coefficients so that we end up with $\mathcal{O}(N_j)$ coefficients. The second compression is defined by additionally setting matrix coefficients to zero which fulfill

$$l' \leq l \qquad \operatorname{dist}(\Omega'_\lambda, \Omega_{\lambda'}) \geq \mathscr{B}'_{l,l'} \tag{19}$$

or

$$l \leq l' \qquad \operatorname{dist}(\Omega_\lambda, \Omega'_{\lambda'}) \geq \mathscr{B}'_{l,l'} \tag{20}$$

with

$$\mathscr{B}'_{l,l'} = a' \max\{2^{-\max\{l,l'\}}, 2^{[j(2d'-r)-\max\{l,l'\}d^*-(l+l')d']/(d^*+r)}\},$$

$$a' > 1$$

and $\Omega'_\lambda = \operatorname{sing\,supp} \psi_\lambda$. Here $\Omega'_\lambda = \operatorname{sing\,supp} \psi_\lambda$ denotes the singular support, i.e. the set of points where the wavelet is not smooth. The result of this criterion is that we can neglect coefficients if one basis function is located in

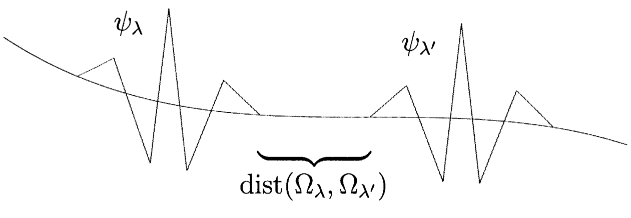

Fig. 6. The situation affected by the first compression.

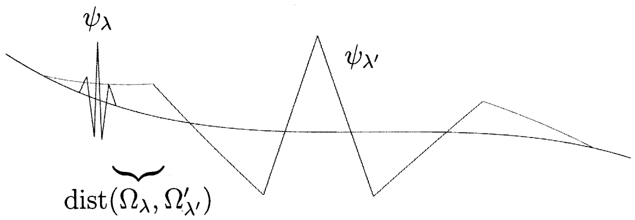

Fig. 7. The situation affected by the second compression.

the support of the other but relatively far away from its singular support (Fig. 7). Clearly, this implies that we get only an effect in such matrix blocks $\mathbf{A}^\varepsilon_{l,l'}$, where the difference of $l$ and $l'$ is large enough, i.e. far away from the diagonal in the blockmatrix $\mathbf{A}^\varepsilon_j$. After the second compression there remain only $\mathcal{O}(N_j)$ entries in the system matrix as the next lemma confirms [12,26,30,40].

**Lemma 5.3.** *Suppose that* $d < d^* + r$, $d'$ *as before and* $\mathrm{dist}(\Omega_\lambda, \Omega_{\lambda'}) \le c2^{-\min\{l,l'\}}$. *After the second compression by neglecting all matrix entries for which*

$$\mathrm{dist}(\Omega'_\lambda, \Omega_{\lambda'}) > \mathscr{B}'_{l,l'}$$

*with*

$$\mathscr{B}'_{l,l'} = a' \max\{2^{-\max\{l,l'\}}, 2^{[j(2d'-r)-\max\{l,l'\}d^*-(l+l')d']/(d^*+r)}\}$$

*the matrix* $\mathbf{A}^\varepsilon_j$ *has only* $\mathcal{O}(N_j)$ *nonzero entries.*

**Remark 5.4.** The second compression requires a large difference between the levels. In practice, we are dealing only with few levels. Nevertheless, the compression criterion works also for matrix coefficients of type $(A\varphi_\lambda, \psi_{\lambda'})$. The only matrix block which cannot be compressed contains the entries of type $(A\varphi_\lambda, \varphi_{\lambda'})$, $|\lambda| = |\lambda'| = 0$. In order to achieve linear complexity we have to guarantee that

$$\dim S_0 \lesssim \sqrt{N_j}.$$

Taking a possibly coarse level on each patch the dimension of $S_0$ is proportional to the number of patches. This gives us a rule of thumb to which extent our method can be applied to complicated geometries. We made the experience that our method works nearly independent of the complexity of the geometry as long as we follow this rule of thumb (Section 8). The adjustment of the parameter $d'$ is similar to that of the parameter $a$.

### *5.3. Setting up the compression pattern*

A naive check of the distance criterion (18) for each matrix coefficient results in an $\mathcal{O}(N_j^2)$-procedure. The following lemma will help us to avoid this and is the basis for an $\mathcal{O}(N_j)$-algorithm for checking the distance criterion.

**Lemma 5.5.** *We consider* $\Omega_{\tilde{\lambda}} \subseteq \Omega_\lambda$ *and* $\Omega_{\tilde{\lambda}'} \subseteq \Omega_{\lambda'}$ *with* $|\tilde{\lambda}| = \tilde{l} \ge l = |\lambda|$ *and* $|\tilde{\lambda}'| = \tilde{l}' \ge l' = |\lambda'|$.

1. *If we assume that*

   $$\mathrm{dist}(\Omega_\lambda, \Omega_{\lambda'}) \ge \mathscr{B}_{l,l'},$$

   *then we obtain that*

   $$\mathrm{dist}(\Omega_{\tilde{\lambda}}, \Omega_{\tilde{\lambda}'}) \ge \mathscr{B}_{\tilde{l},\tilde{l}'}.$$

2. *For* $l \ge l'$ *suppose*

   $$\mathrm{dist}(\Omega_\lambda, \Omega'_{\lambda'}) \ge \mathscr{B}'_{l,l'},$$

   *then we can conclude that*

   $$\mathrm{dist}(\Omega_{\tilde{\lambda}}, \Omega'_{\lambda'}) \ge \mathscr{B}'_{\tilde{l},l'}.$$

With the help of this lemma we only have to check the distance criteria (18)–(20) for coefficients which stem from subdivision of calculated coefficients on a coarser level. In accordance with Lemma 5.3 at most $\mathcal{O}(N_j)$ matrix entries have to be calculated. The resulting procedure of checking the distance criteria is then $\mathcal{O}(N_j)$, too.

## 6. Assembly of the compressed matrix

Up to this point we know that the compressed matrix $\mathbf{A}^\varepsilon_j$ has at most $\mathcal{O}(N_j)$ nonzero entries. Its structure is strictly determined by Eq. (17). Now we have to discuss how to obtain the matrix coefficients

$$(A\psi_{\lambda'}, \psi_\lambda) = \int_{\Omega_\lambda} \int_{\Omega_{\lambda'}} k(x,y)\psi_\lambda(x)\psi_{\lambda'}(y)\mathrm{d}\Gamma_y \,\mathrm{d}\Gamma_x$$

in the Galerkin approach. These coefficients are given by a double integral over the support of the basis functions, which in the case of a three-dimensional problem is a doubled two-dimensional integration. Unfortunately even for cardinal B-splines it is not possible to determine these integrals analytically. Therefore, we are forced to compute the matrix coefficients by numerical integration rules. Numerical integration causes an additional error which has to be controlled and it takes place against a background of realizing asymptotically optimal accuracy while preserving efficiency. This means, the numerical methods have to be chosen carefully, such that the desired linear complexity of the algorithm is not violated. However, it is not obvious that the number of quadrature points employed to compute these $\mathcal{O}(N_j)$ coefficients is still $\mathcal{O}(N_j)$, too. It is an immediate consequence of the fact that we require only a level dependent precision of quadrature [26,40].

**Lemma 6.1.** *Let the error of quadrature for computing the relevant matrix coefficient* $a_{\lambda,\lambda'}$ *be bounded by the level* *dependent accuracy*

$$\epsilon_{l,l'} \sim \min\{2^{-(|l-l'|s)/2}, 2^{-s(j-(l+l')/2)(2d'-r)/(2d^*+r)}\} \times 2^{jr}2^{-2\delta(j-(l+l')/2)} \tag{21}$$

*with $d'$ from the first compression and some $\delta > d$. Then, the Galerkin scheme is stable and converges with the optimal order.*

From Eq. (21) we conclude that the entries on the coarse grids have to be computed with the full accuracy while the entries on the finer grids are allowed to have less accuracy. Unfortunately, the domains of integration are very large on coarser scales.

**Remark 6.2.** We can use Eq. (21) as thresholding parameter improving the a priorly defined compression. Due to our experience such an a posteriori compression improves the rate of compression by a factor of 2–4 [26].

To ensure linear complexity we investigate the number of quadrature points which are permitted for computing the relevant matrix entries with the demanded accuracy. Due to the level dependent precision of quadrature the number of quadrature points is not a constant with respect to the considered level. For this reason we have to pay special attention to counting the total number of quadrature points. As a consequence of matrix compression, the number of elements in the blocks $\mathbf{A}^{\varepsilon}_{l,l'}$ for $l' \gg l$ or $l \gg l'$ is $N^{\alpha}_{l,l'}$ with some $\alpha \in (0,1)$ and $N_{l,l'} \sim 2^{\max\{l,l'\}s}$ the dimension of the block $\mathbf{A}^{\varepsilon}_{l,l'}$ [40]. Therefore, we can use in such matrix blocks $\log(N_{l,l'})$ quadrature points to retain linear complexity for the computation of the matrix $\mathbf{A}^{\varepsilon}_{j}$. A more precise formulation gives us the next theorem.

**Theorem 6.3.** *Suppose $d < d^* + r$, $d'$ as before. Let us further assume that the number of quadrature points $n_{\lambda,\lambda'}$ for the computation of one matrix entry $a_{\lambda,\lambda'}$ is bounded by*

$$n_{\lambda,\lambda'} \le (C_1((j-l)+(j-l'))+C_2)^{\alpha} \tag{22}$$

*for some $\alpha \ge 0$. Then the number of quadrature points for the computation of the compressed matrix is $\mathcal{O}(N_j)$.*

According to the fact that a wavelet is a linear combination of scaling functions, the numerical integration can be reduced to interactions of scaling functions or polynomial functions on certain elements

$$I(\square_{\lambda},\square_{\lambda'}) := \int_{\square_{\lambda}}\int_{\square_{\lambda'}} \mathcal{K}(x,y)p_{\lambda}(x)p_{\lambda'}(y)\mathrm{d}y\,\mathrm{d}x \tag{23}$$

with $\mathcal{K}(x,y)$ defined by Eq. (3). This is quite similar to the traditional Galerkin discretization. The main difference is that in the wavelet approach the elements may appear on different levels due to the multi-level hierarchy of wavelet bases.

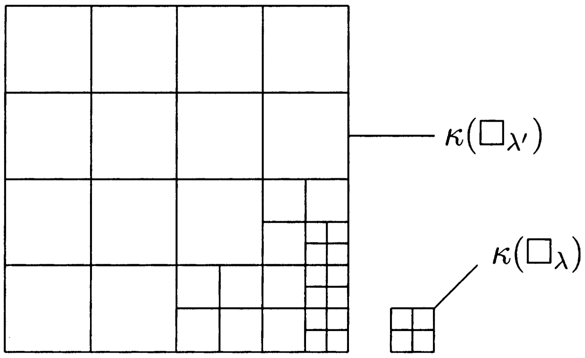

Fig. 8. Adaptive subdivision of the domains of integration with respect to the elements $\kappa(\square_{\lambda})$ and $\kappa(\square_{\lambda'})$.

Difficulties arise if the domains of integration $\kappa(\square_{\lambda})$ and $\kappa(\square_{\lambda'})$ in Eq. (23) are very close together relatively to their size. We have to apply numerical integration carefully in order to keep the number of evaluations of the kernel function at the quadrature knots moderate and to fulfill the assumptions of Theorem 6.3. It is clear that an equidistant subdivision of the domain of integration is not adequate. In Refs. [34,40,41] a geometrically graded subdivision is proposed in combination with varying the polynomial degree of approximation in the integration rules (Fig. 8). It is shown in Refs. [26,30,40] that exponentially convergent quadrature rules combined with such a *hp*-quadrature scheme lead to the number of quadrature points $n_{\lambda,\lambda'}$ satisfying the assumption (22) with $\alpha = 2s$. In practice, tensor product Gauß–Legendre quadrature rules offer exponential convergence.

Since the kernel function has a singularity on the diagonal we are confronted with singular integrals if the domains of integration $\kappa(\square_{\lambda})$ and $\kappa(\square_{\lambda'})$, $l = l'$, have any points in common. We have to give them a special treatment in form of a nonlinear substitution of variable known as the *Duffy transform* [19]. This transform was studied for triangular domains in Ref. [38] and on quadrilaterals in Ref. [39]. Note that singular integrals occur only if the trial and test functions act on the same or on neighbouring patches with a common edge or vertex.

More advanced quadrature techniques limiting the order of integration have been introduced in Ref. [36] for the collocation scheme and in Ref. [35] for the Galerkin scheme. Basis oriented quadrature formulas have been developed in Refs. [1,30].

## 7. Wavelet preconditioning

Let $A : H^{r/2}(\Gamma) \to H^{-r/2}(\Gamma)$ denote a boundary integral operator of order $r \ne 0$. Then, the corresponding system matrix $\mathbf{A}_j$ is ill-conditioned. In fact, there holds

$$\operatorname{cond} \mathbf{A}_j \sim 2^{j|r|}.$$

According to Refs. [11,40], for example, the wavelet approach offers a simple diagonal preconditioner based on the well known norm equivalences of wavelet bases. The preconditioning results from the fact that the wavelets can be normalized in the energy space $H^{r/2}(\Gamma)$ if $\tilde{\gamma} > -r/2$. Here, $\tilde{\gamma}$ denotes the regularity of the *dual* wavelets

$$\tilde{\gamma} = \sup\{q \in \mathbb{R} : \tilde{\psi}_\lambda \in H^q(\Gamma)\}.$$

Let us remark that the regularity of the biorthogonal B-Spline wavelets is well known [43]. Moreover, we mention that this kind of wavelet preconditioning is of additive Schwartz type. For a survey of further preconditioners based on additive Schwartz decompositions, we refer to Ref. [42] and the references therein.

**Theorem 7.1.** *Let the diagonal matrix* $\mathbf{D}_j^q$, $q \in \mathbb{R}$, *be defined by*

$$(\mathbf{D}_j^q)_{\lambda,\lambda'} = 2^{ql}\delta_{\lambda,\lambda'}, \qquad \lambda \in \nabla_l, \;\; \lambda' \in \nabla_{l'}, \;\; -1 \le l, l' < j.$$

*Then, if* $A : H^{r/2}(\Gamma) \to H^{-r/2}(\Gamma)$ *denotes a boundary integral operator of the order* $r$ *with* $\tilde{\gamma} > -r/2$, *the diagonal matrix* $\mathbf{D}_j^r$ *defines a preconditioner to* $\mathbf{A}_j$, *i.e.*

$$\mathrm{cond}(\mathbf{D}_j^{-r/2}\mathbf{A}_j\mathbf{D}_j^{-r/2}) \sim 1.$$

The coefficients on the main diagonal of $\mathbf{A}_j$ satisfy

$$(A\psi_\lambda, \psi_\lambda) \sim 2^{rl}.$$

Therefore, the above preconditioning can be replaced by a diagonal scaling. In fact, the diagonal scaling improves and simplifies the wavelet preconditioning.

**Remark 7.2.** This preconditioning strategy gives uniformly bounded condition numbers which depend on the choice of the wavelet basis. The condition can be relatively large. An advanced preconditioning reducing condition numbers by a magnitude has been introduced in Ref. [26].

## 8. Numerical results

This section is dedicated to numerical examples which confirm our theory. First we solve a Neumann problem employing the indirect formulation for the hypersingular operator. The discretization requires globally continuous piecewise linear wavelets. Second we compute a Dirichlet problem. We use the indirect formulation for the double layer potential operator which gives a Fredholm's integral equation of the second kind. This is approximated by using piecewise constant wavelets. We mention that both problems are chosen such that the solutions are known analytically in order to measure the error of method.

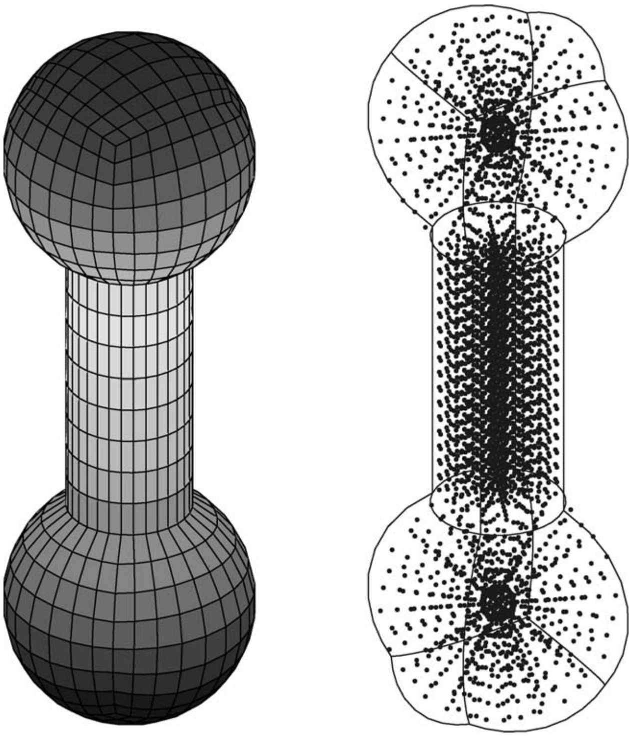

Fig. 9. The mesh on the surface and the evaluation points $\mathbf{x}_i$of the potential.

### 8.1. Neumann problem

For a given $g \in H^{-1/2}(\Gamma)$ with $\int_\Gamma g(x)\mathrm{d}\Gamma_x = 0$ we consider a Neumann problem on the domain $\Omega$, that is, we seek $u \in H^1(\Omega)$ such that

$$\begin{aligned} \Delta u &= 0, \quad \text{in } \Omega, \\ \frac{\partial u}{\partial n} &= g, \quad \text{on } \Gamma. \end{aligned} \tag{24}$$

The considered domain $\Omega$ can be described as the union of two spheres $B_1((0,0,\pm 2)^{\mathrm{T}})$ and one connecting cylinder with the radius 0.5, compare Fig. 9. The boundary $\Gamma$ is represented via 14 patches. Choosing the harmonical function

$$u(x) = \frac{(a, x-b)}{|x-b|^3}, \qquad a = (1,2,4)^{\mathrm{T}}, \;\; b = (1,0,0)^{\mathrm{T}}. \tag{25}$$

Table 1
The maximum norm of the absolute errors of the discrete potential

| Unknowns | | Scaling functions | | Wavelets | |
|---|---|---|---|---|---|
| $j$ | $N_j$ | $\Vert u_j - u_j^{\varphi}\Vert_\infty$ | Contr. | $\Vert u_j - u_j^{\psi}\Vert_\infty$ | Contr. |
| 1 | 58 | 7.1 | – | 7.6 | – |
| 2 | 226 | 4.3 | 1.4 | 4.2 | 1.8 |
| 3 | 898 | 1.2 | 3.6 | 1.2 | 3.5 |
| 4 | 3586 | $1.9 \times 10^{-1}$ | 6.3 | $1.9 \times 10^{-1}$ | 6.2 |
| 5 | 14,338 | $(2.4 \times 10^{-2})$ | (≤8.0) | $1.4 \times 10^{-2}$ | 14 |
| 6 | 57,346 | $(3.0 \times 10^{-3})$ | (≤8.0) | $4.8 \times 10^{-4}$ | 30 |

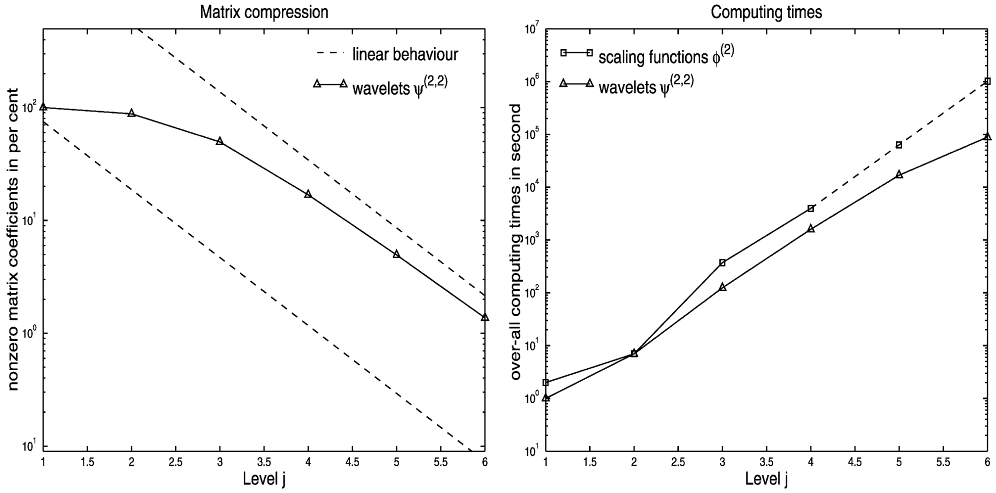

Fig. 10. The compression rates and computing times.

and setting

$$g := \frac{\partial u|_\Gamma}{\partial n}$$

the Neumann problem has the solution $u$ modulo a constant.

The *hypersingular operator* $W$ is given by

$$W\rho(x) := -\frac{1}{4\pi}\frac{\partial}{\partial n_x}\int_\Gamma \frac{\partial}{\partial n_y}\frac{1}{|x-y|}\rho(y)\mathrm{d}\Gamma_y, \qquad x \in \Gamma,$$

and defines an operator of order +1, i.e. $W : H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$. In order to solve problem (24) we seek the density $\rho$ satisfying the Fredholm integral equation of the first kind

$$W\rho = g \qquad \text{on } \Gamma. \tag{26}$$

Since $W$ is symmetric and positive semidefinite [22,30], one restricts $\rho$ by the constraint $\int_\Gamma \rho(x)\mathrm{d}\Gamma_x = 0$. We emphasize that the discretization of the hypersingular operator requires *globally continuous* piecewise linear wavelets since it defines an operator of order +1. According to Lemmas 5.1 and 5.3 piecewise linear wavelets have to provide two vanishing moments.

The density $\rho$ given by the boundary integral equation (26) leads to the solution $u$ of the Neumann problem by

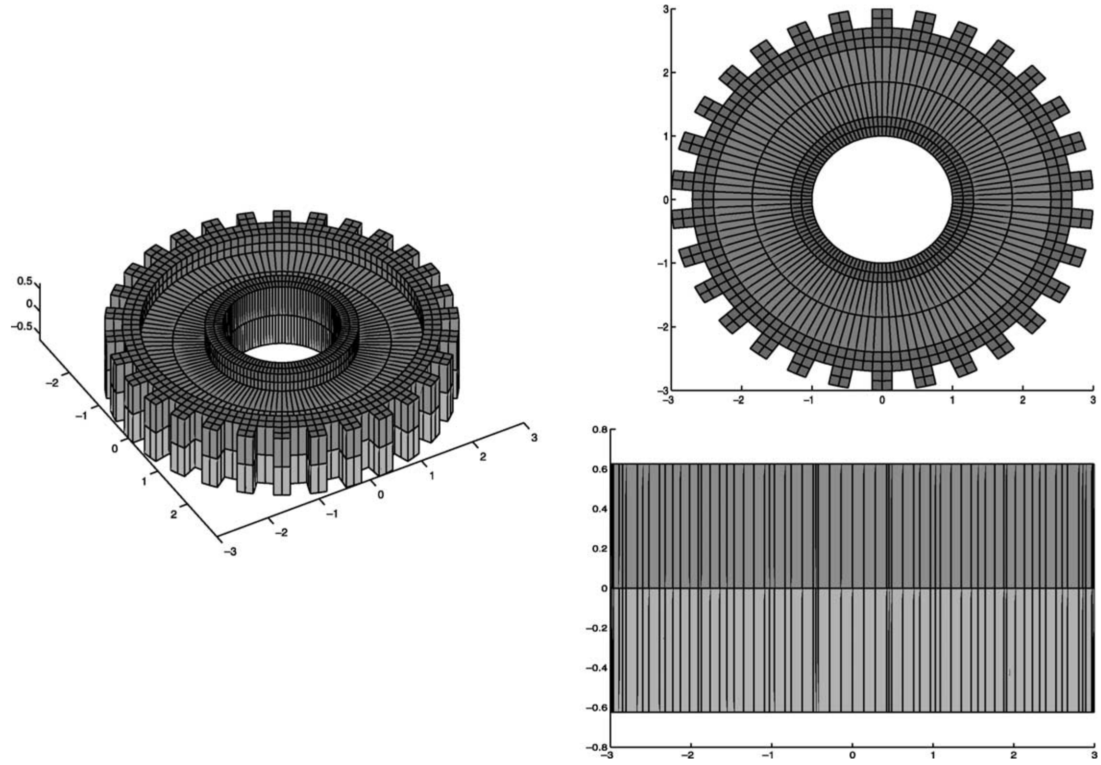

Fig. 11. The description of the mesh of the gearwheel.

Table 2
The maximum norm of the absolute errors of the discrete potential

| Unknowns | | Scaling functions | | Wavelets | |
|---|---|---|---|---|---|
| $j$ | $N_j$ | $\|u_j - u_j^{\varphi}\|_\infty$ | Contr. | $\|u_j - u_j^{\psi}\|_\infty$ | Contr. |
| 1 | 2800 | 0.4 | – | 1.3 | – |
| 2 | 11200 | $1.5\times10^{-1}$ | 2.7 | $1.4\times10^{-1}$ | 9.5 |
| 3 | 44800 | $(3.7\times10^{-2})$ | $(\leq4.0)$ | $4.8\times10^{-2}$ | 2.9 |
| 4 | 179200 | $(9.4\times10^{-3})$ | $(\leq4.0)$ | $1.3\times10^{-2}$ | 3.6 |

application of the double layer potential operator

$$u = K\rho \qquad \text{in } \Omega, \tag{27}$$

cf. Eq. (2). We denote the discrete counterparts by

$$\begin{aligned} &u := (u(x_i)), \qquad u_j^{\varphi} := ((K\rho_j^{\varphi})(x_i)), \\ &u_j^{\psi} := ((K\rho_j^{\psi})(x_i)), \end{aligned} \tag{28}$$

where the evaluation points $x_i$ are specified in Fig. 9. Herein, $u_j^{\varphi}$ indicates the approximation computed by the traditional Galerkin scheme while $u_j^{\psi}$ stands for the numerical solution of the wavelet Galerkin scheme.

First, we compare the errors of approximation with respect to the discrete potentials. The order of convergence is cubic if the density is sufficiently smooth. The columns titled by 'contr.' (contraction) contain the ratio of the absolute error on the previous level and the present error. Optimal convergence means a contraction of 8. As the results in Table 1 confirm, we obtain even a higher rate of convergence. But asymptotically one cannot expect the full order of convergence due to the concave angles between the patches. The wavelet Galerkin scheme achieves the same accuracy as the traditional Galerkin scheme.

In Fig. 10 we visualize the effects of the matrix compression. On the left hand side we plot the number of nonzero coefficients in percent. For 57,346 unknowns the matrix compression yields only 1.37% relevant matrix entries. On the right hand side one figures out the over-all computing times of the traditional discretization compared with those of the fast wavelet discretization. Note that we extrapolated the computing times of the traditional scheme to the levels 5 and 6. On level 6 the speed-up of the wavelet Galerkin scheme is about the factor 11 compared to the traditional scheme.

### 8.2. Dirichlet problem

For a given $f \in H^{1/2}(\Gamma)$ we consider an interior Dirichlet problem, i.e. we seek $u \in H^1(\Omega)$ such that

$$\begin{aligned} \Delta u &= 0, \quad \text{in } \Omega, \\ u &= f, \quad \text{on } \Gamma. \end{aligned} \tag{29}$$

As domain $\Omega$ we consider a gearwheel with 30 teeths (Fig. 11). Let us remark that its surface $\Gamma$ is parametrized via 700 patches. We choose the harmonical potential analogously to Eq. (25) but with $b = (0,0,0)^{\mathrm{T}}$ and set $f := u|_{\Gamma}$. Then, problem (29) has the unique solution $u$.

For solving the Dirichlet problem by the double layer potential operator (2) we apply a Fredholm's integral equation of the second kind

$$(K - \tfrac{1}{2}I)\rho = f, \qquad \text{on } \Gamma. \tag{30}$$

The operator on the left hand side of Eq. (30) defines an operator of order 0, i.e. $K - \frac{1}{2}I : L^2(\Gamma) \to L^2(\Gamma)$. We mention that sometimes it is convenient to take into account $K - \frac{1}{2}I : H^{1/2}(\Gamma) \to H^{1/2}(\Gamma)$ [27]. For the approximation we use piecewise constants. According to Lemmas

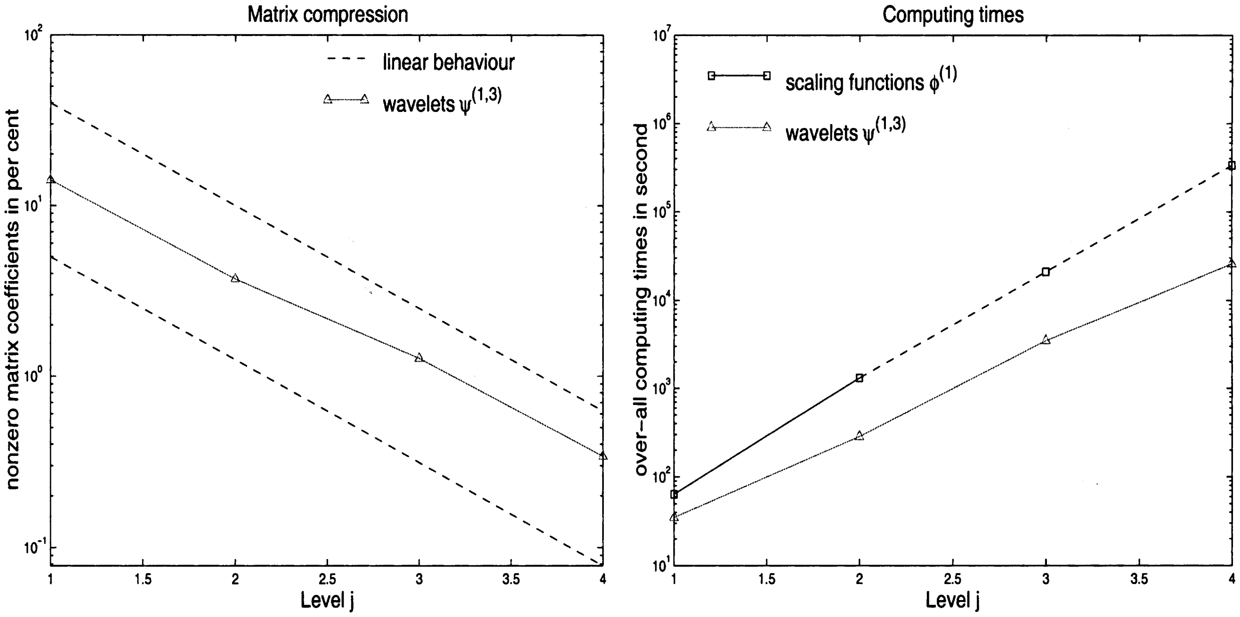

Fig. 12. The compression rates and computing times

5.1 and 5.3 the wavelets must have three vanishing moments. After solving Eq. (30) the solution $u$ is represented as in Eq. (27). The discrete potentials with respect to fixed interior points are denoted according to Eq. (28).

In Table 2 we compare the errors of approximation with respect to the discrete potentials. The order of convergence is quadratic if the density is sufficiently smooth. In this case the contraction is 4. Due to concave angles between the patches this order cannot be expected. But as one figures out the wavelet Galerkin scheme achieves the same accuracy as the traditional Galerkin scheme.

In Fig. 12 the compression rates and computing times are depicted. The number of relevant matrix entries is only 0.34% on the level 4. The traditional scheme would require about 93 h for the computation while the wavelet Galerkin scheme does it in 7 h cpu-time. This means a speed-up of the factor 13.

In order to compare the rates of compression of both computations, we have to take a similar size of the underlying linear systems. Therefore, we consider level 6 ($N_j = 57{,}346$) of the first Example and level 3 ($N_j = 44{,}800$) of the second one. According to Figs. 10 and 12, we observe a similar rate of compression (1.37% respective 1.27% nonzero matrix coefficients) despite the difference between the complexity of the geometries, the applied wavelet bases and the operators under consideration.

## References

[1] Barinka A, Barsch T, Dahlke S, Konik M. Some remarks on quadrature formulas for refinable functions and wavelets, Z Angew Math Mech 2001;81:839–855.

[2] Beylkin G, Coifman R, Rokhlin V. The fast wavelet transform and numerical algorithms. Commun Pure Appl Math 1991;44:141–83.

[3] Canuto C, Tabacco A, Urban K. The wavelet element method. Part I. Construction and analysis. Appl Comput Harm Anal 1999;6:1–52.

[4] Carnicer J, Dahmen W, Peña J. Local decompositions of refinable spaces. Appl Comput Harm Anal 1996;3:127–53.

[5] Chen Z, Micchelli CA, Xu Y. Fast collocation methods for second kind integral equations, SIAM J Numer Anal 2002;40:344–375.

[6] Cohen A, Dahmen W, DeVore R. Adaptive wavelet methods for elliptic operator equations—convergence rates. Math Comput 2001;70:27–75.

[7] Cohen A, Daubechies I, Feauveau J-C. Biorthogonal bases of compactly supported wavelets. Pure Appl Math 1992;45:485–560.

[8] Cohen A, Masson R. Wavelet adaptive method for second order elliptic problems—boundary conditions and domain decomposition. Numer Math 2000;86:193–238.

[9] Costabel M. Principles of boundary element methods. Comput Phys Rep 1987;6:243–74.

[10] Costabel M, Wendland W. Strongly elliptic boundary integral equations. J Reine Angew Math 1986;372:34–63.

[11] Dahmen W. Wavelet and multiscale methods for operator equations. Acta Numer 1997;6:55–228.

[12] Dahmen W, Harbrecht H, Schneider R. Compression techniques for bandary integral equations-optimal complexity estimates. Preprint SFB 393/02-06, TU Chemnitz; 2002, submitted to SIAM J Numer Anal.

[13] Dahmen W, Kunoth A. Multilevel preconditioning. Numer Math 1992;63:315–44.

[14] Dahmen W, Kunoth A, Urban K. Biorthogonal spline-wavelets on the interval—stability and moment conditions. Appl Comput Harm Anal 1999;6:259–302.

[15] Dahmen W, Prößdorf S, Schneider R. Wavelet approximation methods for periodic pseudodifferential equations. Part II. Fast solution and matrix compression. Adv Comput Math 1993;1: 259–335.

[16] Dahmen W, Prößdorf S, Schneider R. In: Chui CK, Montefusco L, Puccio L, editors. Multiscale methods for pseudo-differential equations on smooth manifolds. Proceedings of the International Conference on Wavelets: Theory, Algorithms, and Applications; 1995. p. 385–424.

[17] Dahmen W, Schneider R. Composite wavelet bases for operator equations. Math Comput 1999;68:1533–67.

[18] Dahmen W, Stevenson R. Element-by-element construction of wavelets satisfying stability and moment conditions. SIAM J Numer Anal 1999;37:319–52.

[19] Duffy M. Quadrature over a pyramid or cube of integrands with a singularity at the vertex. SIAM J Numer Anal 1982;19:1260–2.

[20] Goreinov S, Tyrtischnikov E, Yeremin Y. Matrix-free iterative solution strategies for large dense linear systems. Numer Linear Algebra Appl 1997;4:273–94.

[21] Greengard L, Rokhlin V. A fast algorithm for particle simulation. J Comput Phys 1987;73:325–48.

[22] Hackbusch W. Integralgleichungen. Stuttgart: B.G. Teubner; 1989.

[23] Hackbusch W. A sparse matrix arithmetic based on $\mathscr{H}$-matrices. Part I. Introduction to $\mathscr{H}$-matrices. Computing 1999;64:89–108.

[24] Hackbusch W, Khoromskij B, Sauter S. On $\mathscr{H}_2$ matrices. In: Bungatz H-J, Hoppe R, Zenger Z, editors. Lectures on applied mathematics. Heidelberg: Springer; 2000. p. 9–30.

[25] Hackbusch W, Nowak ZP. On the fast matrix multiplication in the boundary element method by panel clustering. Numer Math 1989;54: 463–91.

[26] Harbrecht H. Wavelet Galerkin schemes for the boundary element method in three dimensions. PhD Thesis. Technische Universität Chemnitz, Germany; 2001.

[27] Harbrecht H, Paiva F, Pérez C, Schneider R. Biorthogonal wavelet approximation for the coupling of FEM–BEM. Numer Math 2002;92: 325–356.

[28] Höllig K, Mögerle H. G-splines. Comput-Aid Geom Des 1990;7: 197–207.

[29] Hsiao G, Wendland W. A finite element method for some integral equations of the first kind. J Math Anal Appl 1977;58(3): 449–81.

[30] Konik M. A fully discrete wavelet Galerkin boundary element method in three dimensions. PhD Thesis. Technische Universität Chemnitz, Germany; 2001.

[31] Lage C, Schwab C. Wavelet Galerkin algorithms for boundary integral equations. SIAM J Sci Comput 1999;20:2195–222.

[32] von Petersdorff T, Schneider R, Schwab C. Multiwavelets for second kind integral equations. SIAM J Numer Anal 1997;34:2212–27.

[33] von Petersdorff T, Schwab C. Wavelet approximation for first kind integral equations on polygons. Numer Anal 1996;74:479–516.

[34] von Petersdorff T, Schwab C. Fully discretized multiscale Galerkin BEM. In: Dahmen W, Kurdila A, Oswald P, editors. Multiscale wavelet methods for PDEs. San Diego: Academic Press; 1997. p. 287–346.

[35] Rathsfeld A. A quadrature algorithm for wavelet Galerkin methods. Über Waveletalgorithmen für die Randelementmethode. Habilitation Thesis. Technische Universität Chemnitz, Germany; 2001.

[36] Rathsfeld A, Schneider R. On a quadrature algorithm for the piecewise linear collocation applied to boundary integral equations. Preprint SFB 393/00-15, TU Chemnitz; 2000, to appear in Math Mech in the Appl Sci.

[37] Reif U. Biquadratic G-spline surfaces. Comput-Aid Geom Des 1995;12:193–205.

[38] Sauter S. Über die effiziente Verwendung des Galerkin Verfahrens zur Lösung Fredholmscher Integralgleichungen. PhD Thesis. Christian-Albrecht-Universität Kiel, Germany; 1992.

[39] Sauter S, Schwab C. Quadrature for the *hp*-Galerkin BEM in $\mathbb{R}^3$. Numer Math 1997;78:211–58.

[40] Schneider R. Multiskalen- und Wavelet-Matrixkompression: analysisbasierte Methoden zur Lösung großer vollbesetzter Gleichungssysteme. Stuttgart: B.G. Teubner; 1998.

[41] Schwab C. Variable order composite quadrature of singular and nearly singular integrals. Computing 1994;53:173–94.

[42] Stephan EP. Multilevel methods for the *h*-, *p*- and *hp*-versions of the boundary element method. J Comput Appl Math 2000;125:503–19.

[43] Villemoes L. Wavelet analysis of refinement equations. SIAM J Math Anal 1994;25:1433–60.

[44] Wendland W. Boundary element methods and their asymptotic convergence. In: Filippi P, editor. Theoretical acoustics and numerical techniques. CSIM Courses 277, Wien-New York: Springer; 1983. p. 135–216.