SIAM J. SCI. COMPUT.
Vol. 27, No. 4, pp. 1347–1370

WAVELET GALERKIN SCHEMES FOR BOUNDARY INTEGRAL EQUATIONS—IMPLEMENTATION AND QUADRATURE*

HELMUT HARBRECHT† AND REINHOLD SCHNEIDER†

Abstract. In the present paper we consider the fully discrete wavelet Galerkin scheme for the fast solution of boundary integral equations in three dimensions. It produces approximate solutions within discretization error accuracy offered by the underlying Galerkin method at a computational expense that stays proportional to the number of unknowns. We focus on algorithmical details of the scheme, in particular on numerical integration of relevant matrix coefficients. We illustrate the proposed algorithm by numerical results.

Key words. boundary integral equations, biorthogonal wavelet bases, matrix compression, numerical integration

AMS subject classifications. 47A20, 65F10, 65F50, 65N38, 65R20

DOI. 10.1137/S1064827503429387

Introduction. Various problems in science and engineering can be formulated as boundary integral equations. In general, boundary integral equations are solved numerically by the boundary element method (BEM). For example, BEM is a favorable approach for the treatment of exterior boundary value problems. Nevertheless, traditional discretizations of integral equations suffer from a major disadvantage. The associated system matrices are densely populated. Therefore, the complexity for solving such equations is at least $\mathcal{O}(N_J^2)$, where N_J denotes the number of equations. This fact seriously restricts the maximal size of the linear equations.

Modern methods for the fast solution of BEM reduce the complexity to a suboptimal rate $\mathcal{O}(N_J \log^\alpha N_J)$ or even an optimal rate $\mathcal{O}(N_J)$. Prominent examples for such methods are the *fast multipole method* [16], the *panel clustering* [18], *$\mathcal{H}$-matrices* [17], or the *wavelet Galerkin scheme* [1, 6, 10, 11, 31]. In fact, a Galerkin discretization with wavelet bases results in quasi-sparse matrices; i.e., most matrix entries are negligible and can be treated as zero. Discarding these nonrelevant matrix entries is called matrix compression. As shown in [6, 31] only $\mathcal{O}(N_J)$ significant matrix entries remain.

Concerning boundary integral equations, a strong effort has been expended on the construction of appropriate wavelet bases on surfaces [8, 12, 13, 14, 20, 26]. In order to achieve the optimal complexity of the wavelet Galerkin scheme, wavelet bases are required which provide a sufficiently large number of vanishing moments. Our realization is based on biorthogonal spline wavelets derived from the multiresolution developed in [3]. These wavelets are advantageous since the regularity of the duals is known [33]. Moreover, the duals are compactly supported, which preserves the linear complexity of the fast wavelet transform also for its inverse. This is an important task for the coupling of the finite element method and BEM; cf. [21, 22]. Additionally,

*Received by the editors June 11, 2003; accepted for publication (in revised form) June 8, 2005; published electronically January 6, 2006. This work is supported in part by the European Community's Human Potential Programme under contract HPRN-CT-202-00286 (BREAKING COMPLEXITY) and the SFB 393, "Numerical Simulation on Massive Parallel Computers," funded by the Deutsche Forschungsgemeinschaft.

http://www.siam.org/journals/sisc/27-4/42938.html

†Institute of Computer Science and Applied Mathematics, Christian-Albrechts-University of Kiel, Olshausenstr. 40, 24098 Kiel, Gemany (hh@numerik.uni-kiel.de, rs@numerik.uni-kiel.de).

in view of the discretization of operators of positive order, e.g., the hypersingular operator, globally continuous wavelets are available [2, 4, 12, 20]. We refer the reader to [23] for the details on the wavelet bases employed for our actual implementation of the wavelet Galerkin scheme.

In the present paper we describe algorithmical details of a fully discrete wavelet Galerkin scheme for the fast solution of boundary integral equations. Particularly, we realized the so-called second compression to achieve linear complexity [6, 31]. The efficient computation of the relevant matrix coefficients turned out to be the cornerstone for the successful application of the wavelet Galerkin method [20, 27, 31]. A fully discrete Galerkin scheme based on numerical quadrature is also required to treat nontrivial geometries. We expended much effort to realize an efficient quadrature strategy. Supposing that the given manifold is piecewise analytic, one applies an *hp*-quadrature scheme [20, 27, 31, 32] based on tensor product Gauss–Legendre quadrature formulas to arrive at an algorithm with asymptotically linear complexity without compromising the accuracy of the Galerkin scheme. However, a careful organization of the quadrature is indispensable to avoid wasteful and multiple calculations that affect the performance. We propose an element-based quadrature similar to the traditional single-scale Galerkin scheme. Especially we require only the quadrature routines of the traditional scheme to compute element-element interactions, except for the extension to different degrees of quadrature to deal with elements on different levels. The *h*-refinement is realized by replacing an element-element interaction by the element-element interactions related to the sons of the elements. Combined with a *recycling strategy* to reuse previously computed element-element interactions (cf. sections 8 and 9) the work for quadrature decreases significantly. We emphasize that the proposed strategy greatly reduces the break-even point. By numerical experiments, considering the sphere and a nontrivial gearwheel, we confirm that we proceeded in developing a fast solver for the considered class of problems.

The outline of the present paper is as follows. First, in section 1, we introduce the class of problems under consideration. Then we provide the wavelet bases on manifolds in section 2 and recall the matrix compression in section 3. The next sections are concerned with implementational aspects: In section 4 a suitable data structure is introduced to handle a wide class of wavelets, in section 5 the numerical evaluation of distances is performed, in section 6 the compression pattern is set up, and in section 7 the computation of the compressed system matrix is reduced to the computation of certain element-element interactions which are investigated in section 8. Section 9 is concerned with the quadrature of element-element interactions which correspond mostly to nearly singular and singular integrals. We prove that, using our quadrature strategy, the compressed system matrix is computed within linear complexity. In section 10 we present numerical results employing both piecewise constant and bilinear wavelets.

We shall frequently write $a \lesssim b$ to express that a is bounded by a constant multiple of b, uniformly with respect to all parameters on which a and b may depend. Then $a \sim b$ means $a \lesssim b$ and $b \lesssim a$.

1. Problem formulation and preliminaries. We consider boundary integral equations on a closed boundary surface Γ of a three-dimensional domain $\Omega \subset \mathbb{R}^3$,

$$Au = f \quad \text{on } \Gamma, \tag{1.1}$$

where the boundary integral operator

$$Au(x) = \int_\Gamma k(x,y)u(y)d\Gamma_y$$

is assumed to be an operator of order $2q$, that is, $A : H^q(\Gamma) \to H^{-q}(\Gamma)$. The kernel functions under consideration are supposed to be piecewise smooth as functions in the variables x, y, apart from the diagonal $\{(x, y) \in \Gamma \times \Gamma : x = y\}$, and may have a singularity on the diagonal. Such kernels arise, for instance, by applying a boundary integral formulation to a second order elliptic problem. In general, they decay like a negative power of the distance of the arguments that depends on the operator order $2q$.

Throughout the remainder of this paper we shall assume that the boundary manifold Γ is given as a parametric surface consisting of smooth patches. More precisely, let $\square := [0, 1]^2$ denote the unit square. The manifold $\Gamma \in \mathbb{R}^3$ is partitioned into a finite number of *patches*

$$\Gamma = \bigcup_{i=1}^{M} \Gamma_i, \qquad \Gamma_i = \gamma_i(\square), \qquad i = 1, 2, \ldots, M, \tag{1.2}$$

where each $\gamma_i : \square \to \Gamma_i$ defines a diffeomorphism of $\square$ onto Γ_i. The intersection $\Gamma_i \cap \Gamma_{i'}$, $i \neq i'$, of two different patches is supposed to be either empty or a common edge or vertex.

A mesh of level j on Γ is induced by dyadic subdivisions of depth j of the unit square into 4^j squares

$$\square_{j,k} := [2^{-j}k_1, 2^{-j}(k_1 + 1)] \times [2^{-j}k_2, 2^{-j}(k_2 + 1)] \subseteq \square,$$

where $k = (k_1, k_2)$ with $0 \leq k_1, k_2 < 2^j$. This generates $4^j M$ *elements* (or elementary domains) $\Gamma_{i,j,k} := \gamma_i(\square_{j,k}) \subseteq \Gamma_i$, $i = 1, \ldots, M$.

In order to ensure that the collection of elements $\{\Gamma_{i,j,k}\}$ on the level j forms a regular mesh on Γ, the parametric representation is subjected to the following *matching condition*: A bijective, affine mapping $\Xi : \square \to \square$ exists such that for all $x = \gamma_i(s)$ on a common edge of Γ_i and $\Gamma_{i'}$ it holds that $\gamma_i(s) = (\gamma_{i'} \circ \Xi)(s)$. In other words, the diffeomorphisms γ_i and $\gamma_{i'}$ yield the same parametrization of the common edge except for orientation. This setting is frequently used and well understood in computer aided geometric design; cf. [24, 28], for example. A conforming mesh of a gearwheel is depicted in Figure 1.1.

Denoting the surface measure by $\kappa_i(s) := \|\partial_{s_1}\gamma_i(s) \times \partial_{s_2}\gamma_i(s)\|$, the canonical inner product in $L^2(\Gamma)$ is given by

$$\langle u, v\rangle := \int_\Gamma u(x)v(x)d\Gamma_x = \sum_{i=1}^{M} \int_\square u\big(\gamma_i(s)\big)v\big(\gamma_i(s)\big)\kappa_i(s)ds.$$

The associated Sobolev spaces are denoted by $H^t(\Gamma)$, endowed with the norms $\|\cdot\|_t$, where for $t < 0$ it is understood that $H^t(\Gamma) = (H^{-t}(\Gamma))'$. Of course, depending on the global smoothness of the surface, the range of permitted $t \in \mathbb{R}$ is limited to $t \in (-t_\Gamma, t_\Gamma)$.

In what follows it will be rather convenient to have access to the *local parametrizations* $\gamma_{i,j,k} : \square \to \Gamma_{i,j,k}$ given by

$$\gamma_{i,j,k}(s) := \gamma_i\big(2^{-j}\left[\begin{smallmatrix} k_1+s_1 \\ k_2+s_2 \end{smallmatrix}\right]\big), \qquad s = \left[\begin{smallmatrix} s_1 \\ s_2 \end{smallmatrix}\right] \in \square. \tag{1.3}$$

For $m \in \mathbb{N}$ we conclude that

$$\partial_{s_n}^m \gamma_{i,j,k}(s) = 2^{-jm}\ (\partial_{s_n}^m \gamma_i)\big(2^{-j}\left[\begin{smallmatrix} k_1+s_1 \\ k_2+s_2 \end{smallmatrix}\right]\big), \qquad n = 1, 2; \tag{1.4}$$

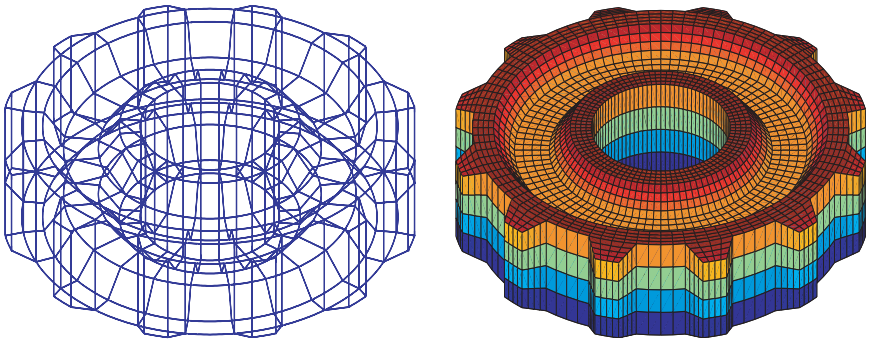

FIG. 1.1. *A gearwheel represented by* 336 *patches and the associated mesh on the level* 2.

in particular, the corresponding surface measure satisfies

$$\kappa_{i,j,k}(s) := \big\|\partial_{s_1}\gamma_{i,j,k}(s) \times \partial_{s_2}\gamma_{i,j,k}(s)\big\| = 2^{-2j}\kappa_i\big(2^{-j}\left[\begin{smallmatrix}k_1+s_1\\ k_2+s_2\end{smallmatrix}\right]\big). \tag{1.5}$$

We can now specify the kernel functions. To this end, we denote by $\alpha = (\alpha_1, \alpha_2)$ and $\beta = (\beta_1, \beta_2)$ multi-indices of dimension two and define $|\alpha| := \alpha_1 + \alpha_2$.

DEFINITION 1.1. *A kernel* $k(x, y)$ *is called a standard kernel of order* $2q$ *if the partial derivatives of the transported kernel functions*

$$k_{i,i'}(s,t) := k\big(\gamma_i(s), \gamma_{i'}(t)\big)\kappa_i(s)\kappa_{i'}(t), \qquad s,t \in \square, \qquad 1 \le i, i' \le M, \tag{1.6}$$

are bounded by

$$|\partial_s^\alpha \partial_t^\beta k_{i,i'}(s,t)| \le c_{\alpha,\beta}\|\gamma_i(s) - \gamma_{i'}(t)\|^{-(2+2q+|\alpha|+|\beta|)},$$

provided that $2 + 2q + |\alpha| + |\beta| > 0$.

We emphasize that this definition requires patchwise smoothness but *not* global smoothness of the geometry. The surface itself needs to be only Lipschitz. Generally, under this assumption, the kernel of a boundary integral operator A of order $2q$ is a standard kernel of order $2q$. Hence, we may assume this property in what follows. We shall encounter further specifications in connection with numerical integration.

2. Biorthogonal wavelet bases. The nested trial spaces $V_j \subset V_{j+1}$ that we shall employ in the Galerkin scheme are the spaces of piecewise constant or bilinear functions on the given partition. These trial spaces have the *approximation order* $d = 1$ and $d = 2$ in the case of the piecewise constants and bilinears, respectively; i.e., $\inf_{v_j \in V_j} \|v - v_j\|_0 \lesssim 2^{-jd}\|v\|_d$. They are spanned by so-called *single-scale bases* $\Phi_j = \{\phi_{j,k} : k \in \Delta_j\}$ which can be specified as follows: On the level j, we find for each element $\Gamma_{i',j,k'}$ a piecewise constant scaling function $\phi_{j,k}$ with

$$\phi_{j,k}\big|_{\Gamma_{i',j,k'}} \equiv 2^j \tag{2.1}$$

and $\phi_{j,k}(x) = 0$ elsewhere. To describe the canonical piecewise bilinear scaling functions, we define four bilinear shape functions on the unit square:

$$\begin{aligned} p_1^\square(s) &:= (1-s_1)(1-s_2), & p_2^\square(s) &:= s_1(1-s_2),\\ p_3^\square(s) &:= s_1 s_2, & p_4^\square(s) &:= (1-s_1)s_2. \end{aligned} \tag{2.2}$$

Thus, $\phi_{j,k}$ is equal to 2^j in one node and equal to zero in the remaining nodes and, if its support contains an element $\Gamma_{i',j,k'}$, we find an $m \in \{1,2,3,4\}$ such that

$$\phi_{j,k}\big|_{\Gamma_{i',j,k'}}(x) = 2^j p_m^{\square}(s), \qquad x = \gamma_{i',j,k'}(s) \in \Gamma_{i',j,k'}. \tag{2.3}$$

Continuity is supposed on each patch, but along the interfaces of the patches we may consider double nodes or continuity. Notice that our definition yields the L^2-normalization $\|\phi_{j,k}\|_0 \sim 1$.

Associated with the *multiresolution sequence* $\{V_j\}_{j\geq j_0}$ is always a *dual* multiresolution sequence $\{\widetilde{V}_j\}_{j\geq j_0}$ which is generated by dual bases $\widetilde{\Phi}_j = \{\widetilde{\phi}_{j,k} : k \in \Delta_j\}$; i.e., one has $\langle \phi_{j,k}, \widetilde{\phi}_{j,k'}\rangle = \delta_{k,k'}$, $k,k' \in \Delta_j$. Here and below j_0 always stands for some fixed coarsest level of resolution that may depend on Γ. For the current type of boundary surfaces Γ the $\Phi_j, \widetilde{\Phi}_j$ are generated by first constructing dual pairs of single-scale bases on the interval $[0,1]$, using the dual components from [3] adapted to the interval [9]. Tensor products yield corresponding dual pairs on $\square$. Using the parametric liftings γ_i and gluing across patch boundaries leads to globally continuous single-scale bases $\Phi_j, \widetilde{\Phi}_j$ on Γ [2, 4, 13]. The dual spaces have approximation order $\widetilde{d} \geq d$ such that $d + \widetilde{d}$ is even.

Given the single-scale bases $\Phi_j, \widetilde{\Phi}_j$, one now can construct biorthogonal *complement bases* $\Psi_j = \{\psi_{j,k} : k \in \nabla_j := \triangle_{j+1} \setminus \triangle_j\}$, $\widetilde{\Psi}_j = \{\widetilde{\psi}_{j,k} : k \in \nabla_j\}$, i.e., $\langle \psi_{j,k}, \widetilde{\psi}_{j',k'}\rangle = \delta_{(j,k),(j',k')}$, such that

$$\begin{aligned} W_j &:= \operatorname{span}\{\Psi_j\} \perp \widetilde{V}_j, \qquad & V_{j+1} &= V_j \oplus W_j, \\ \widetilde{W}_j &:= \operatorname{span}\{\widetilde{\Psi}_j\} \perp V_j, \qquad & \widetilde{V}_{j+1} &= \widetilde{V}_j \oplus \widetilde{W}_j, \end{aligned}$$

and $\operatorname{diam}\operatorname{supp}\psi_{j,k} \sim \operatorname{diam}\operatorname{supp}\widetilde{\psi}_{j,k} \sim 2^{-j}$ $(j \geq j_0)$; see, e.g., [2, 4, 12, 13] and [20, 23] for particularly useful local representations of important construction ingredients. We suppose these complement bases are normalized in $L^2(\Gamma)$.

A biorthogonal or *dual* pair of wavelet bases is now obtained by taking the coarse single-scale basis and the union of the complement bases,

$$\Psi = \bigcup_{j\geq j_0-1} \Psi_j, \qquad \widetilde{\Psi} = \bigcup_{j\geq j_0-1} \widetilde{\Psi}_j,$$

where we have set for convenience $\Psi_{j_0-1} := \Phi_{j_0}$, $\widetilde{\Psi}_{j_0-1} := \widetilde{\Phi}_{j_0}$. Of course, in the infinite-dimensional case the notion of basis has to be made more specific. The key feature of the wavelet basis is now the fact that $\Psi, \widetilde{\Psi}$ are actually *Riesz bases* in $L^2(\Gamma)$.

From biorthogonality and the fact that the dual spaces $\widetilde{V}_j$ have the approximation order $\widetilde{d}$ one infers *vanishing moments* or the *cancellation property* of the primal wavelets

$$|\langle v, \psi_{j,k}\rangle| \lesssim 2^{-j(\tilde{d}+1)} |v|_{W^{\tilde{d},\infty}(\operatorname{supp}\psi_{j,k})}. \tag{2.4}$$

Here $|v|_{W^{\tilde{d},\infty}(\Omega)} := \sup_{|\alpha|=\tilde{d},\, x\in\Omega} |\partial^\alpha v(x)|$ denotes the seminorm in $W^{\tilde{d},\infty}(\Omega)$. The fact that the concept of biorthogonality allows us to choose the order $\widetilde{d}$ of vanishing moments higher than the approximation order d is essential for deriving optimal compression strategies that could not be realized by orthonormal bases.

3. Matrix compression. We shall be concerned with the Galerkin method for the solution of the given boundary integral equation (1.1): Find $u_J \in V_J$ solving the variational problem

$$\langle Au_J, v_J\rangle = \langle f, v_J\rangle \quad \text{for all} \quad v_J \in V_J.$$

Traditionally this equation is discretized by the single-scale basis of V_J which yields a densely populated system matrix. Using instead wavelets with a sufficiently strong cancellation property (2.4), the system matrix becomes quasi-sparse and most matrix coefficients are negligible without compromising the order of convergence of the Galerkin scheme [6, 31].

But before we formulate this result, we introduce the following abbreviation:

$$\Omega_{j,k} := \operatorname{conv\,hull}(\operatorname{supp}\psi_{j,k}), \qquad \Omega'_{j,k} := \operatorname{sing\,supp}\psi_{j,k}. \tag{3.1}$$

Notice that the first expression denotes the convex hull of the support of a wavelet with respect to the Euclidean space $\mathbb{R}^3$. The second expression indicates the *singular support*, i.e., that subset of Γ where the wavelet is not smooth.

THEOREM 3.1 (a priori compression). *Let $\Omega_{j,k}$ and $\Omega'_{j,k}$ be given as in* (3.1) *and define the compressed system matrix $\mathbf{A}_J$, corresponding to the boundary integral operator A, by*

$$[\mathbf{A}_J]_{(j,k),(j',k')} := \begin{cases} 0, & \operatorname{dist}(\Omega_{j,k},\Omega_{j',k'}) > \mathcal{B}_{j,j'} \text{ and } j,j' \geq j_0, \\ 0, & \operatorname{dist}(\Omega_{j,k},\Omega_{j',k'}) \lesssim 2^{-\min\{j,j'\}} \text{ and} \\ & \operatorname{dist}(\Omega'_{j,k},\Omega_{j',k'}) > \mathcal{B}'_{j,j} \text{ if } j' > j \geq j_0 - 1, \\ & \operatorname{dist}(\Omega_{j,k},\Omega'_{j',k'}) > \mathcal{B}'_{j,j} \text{ if } j > j' \geq j_0 - 1, \\ \langle A\psi_{j',k'},\psi_{j,k}\rangle & \text{otherwise.} \end{cases} \tag{3.2}$$

Fixing

$$a, a' > 1, \qquad d < d' < \widetilde{d} + 2q, \tag{3.3}$$

the cut-off parameters $\mathcal{B}_{j,j'}$ and $\mathcal{B}'_{j,j'}$ are set as follows:

$$\begin{aligned} \mathcal{B}_{j,j'} &= a \ \max\left\{2^{-\min\{j,j'\}}, 2^{\frac{2J(d'-q)-(j+j')(d'+\tilde{d})}{2(\tilde{d}+q)}}\right\}, \\ \mathcal{B}'_{j,j'} &= a' \max\left\{2^{-\max\{j,j'\}}, 2^{\frac{2J(d'-q)-(j+j')d'-\max\{j,j'\}\tilde{d}}{\tilde{d}+2q}}\right\}. \end{aligned} \tag{3.4}$$

Then, the error estimate

$$\|u - u_J\|_{2q-d} \lesssim 2^{-2J(d-q)}\|u\|_d \tag{3.5}$$

holds for the solution u_J of the compressed Galerkin system provided that u and Γ are sufficiently regular.

In [6, 20] we presented a general theorem which shows that the overall complexity of assembling the compressed system matrix can be kept of the order $\mathcal{O}(N_J)$, $N_J = \dim V_J$, even when a computational cost of logarithmic order is allowed for each entry. This theorem will be used later as the essential ingredient in proving that the quadrature strategy proposed in section 9 scales linearly.

THEOREM 3.2. *Assume that* $\mathbf{A}_J$ *is compressed according to* (3.2). *The complexity of computing this compressed matrix is* $\mathcal{O}(N_J)$ *provided that for some* $\alpha \geq 0$ *at most* $\mathcal{O}\big([2J-j-j']^4\big)$ *operations are spent on the approximate calculation of the nonvanishing entries* $\langle A\psi_{j',k'}, \psi_{j,k}\rangle$.

Numerical integration of relevant matrix coefficients has to be performed with sufficient accuracy. It turns out that the higher the level of resolution the lower the required accuracy. In accordance with [6, 20] we can formulate the following sufficient condition to retain the optimal order of convergence of the Galerkin scheme.

THEOREM 3.3. *Let the matrix* $\mathbf{A}_J$ *denote the compressed system matrix according to Theorem* 3.1 *and consider the perturbed system matrix* $\widetilde{\mathbf{A}}_J$ *satisfying*

$$\big|[\mathbf{A}_J - \widetilde{\mathbf{A}}_J]_{(j,k),(j',k')}\big| \leq \varepsilon_{j,j'}$$

for all relevant coefficients, where the level dependent error $\varepsilon_{j,j'}$ *is given by*

$$\varepsilon_{j,j'} \sim \min\Big\{2^{-|j-j'|}, 2^{-(2J-j-j')\frac{d'-q}{d+q}}\Big\}2^{2Jq}2^{-\delta(2J-j-j')} \tag{3.6}$$

with $d' \in (d, \widetilde{d}+2q)$ *from* (3.3) *and* $\delta > d$. *Then, the solution* u_J *of the perturbed Galerkin system satisfies* (3.5) *provided that* u *and* Γ *are sufficiently regular.*

If the order q of the boundary integral operator A is $\neq 0$, the compressed system matrix $\mathbf{A}_J$ becomes more and more ill conditioned when J increases. However, as a consequence of the norm equivalences of wavelet bases, the diagonally scaled system matrix has uniformly bounded spectral condition numbers, provided that the regularity $\widetilde{\gamma}$ of the dual wavelets satisfies $\widetilde{\gamma} > -q$ [5, 7, 10, 31]. Let us further remark that based on (3.6) an a posteriori compression is also available, which reduces again the number of nonzero coefficients; cf. [6, 20].

4. The data structure. In general the wavelet bases are not simply defined via tensor products of the univariate case. In particular, considering globally continuous wavelets, a single wavelet might be supported on several patches. This requires a suitable data structure to handle such wavelet bases.

4.1. The element tree. According to section 1 the elements $\Gamma_{i,j,k} = \gamma_i(\square_{j,k})$ are the images of the refined unit square under the parametric liftings γ_i. We introduce a hierarchical element tree with respect to "$\subseteq$" as follows. Starting with the M patches $\Gamma_i := \Gamma_{i,0,(0,0)}$ as the first generation, each element $\Gamma_{i,j,k}$ has the four sons $\Gamma_{i,j+1,k'} \subseteq \Gamma_{i,j,k}$, $k' \in \big\{(2k_1, 2k_2), (2k_1+1, 2k_2), (2k_1+1, 2k_2+1), (2k_1, 2k_2+1)\big\}$. Denoting by J the level of discretization, the tree consists of $J+1$ generations.

Notice that this element tree is given implicitly. The number of all elements is given by $N_\Gamma := \frac{M}{3}(4^{J+1}-1)$. Hence, we may utilize the bijective mapping

$$\lambda := \theta(i,j,k) = \frac{M}{3}(4^j - 1) + 4^j M i + 2^j k_2 + k_1 \tag{4.1}$$

to indicate the element $\Gamma_{i,j,k}$ uniquely by an integer $\lambda \in \{0, 1, \ldots, N_\Gamma - 1\}$. This numbering corresponds to indexing the element tree line-by-line. For the sake of brevity we frequently will write Γ_λ instead of $\Gamma_{i,j,k}$. Moreover, it is rather convenient to set $|\lambda| := j$.

4.2. The wavelet representation. The common way to perform the quadrature for, e.g., piecewise bilinear functions is element-based; that is, for all pairs of

elements one computes the element-element interactions of the associated shape functions and updates the system matrix correspondingly. Roughly speaking, a basis function is represented by in general four quadrilaterals (we call them the elements) and certain weights (namely equal to 2^j in the common vertex and zero elsewhere) of the shape functions associated with the vertices of these quadrilaterals. Since the wavelets of the level j are contained in V_{j+1}, we can represent the wavelet $\psi_{j,k}$ similarly by a set of elements $\Gamma_{i',j+1,k'} \subset \operatorname{supp} \psi_{j,k}$ and certain weight factors of the associated shape functions. This enables us to perform an element-based quadrature. It turns out that parts of the support of the wavelets are smooth even on the previous level, i.e., with respect to the elements $\Gamma_{i',j,k'}$; cf. [20, 23], for example. Therefore, we recommend a two level representation of the wavelets, which increases the performance of the wavelet Galerkin scheme enormously since fewer element-element interactions have to be evaluated.

More precisely, for each wavelet there exists a set $\mathcal{L}_{j,k}$ of elements $\Gamma_\lambda \in \operatorname{supp} \psi_{j,k}$, $|\lambda| \in \{j, j+1\}$, satisfying

$$\operatorname{supp} \psi_{j,k} = \bigcup_{\Gamma_\lambda \in \mathcal{L}_{j,k}} \Gamma_\lambda \subseteq \Omega_{j,k}, \qquad \Omega'_{j,k} \subseteq \bigcup_{\Gamma_\lambda \in \mathcal{L}_{j,k}} \partial\Gamma_\lambda \tag{4.2}$$

such that $\psi_{j,k} \circ \gamma_\lambda$ is either constant ($d = 1$),

$$\psi_{j,k}(x) \equiv 2^{|\lambda|} \omega_{j,k,\lambda}, \quad x = \gamma_\lambda(s), \quad s \in \square, \quad \Gamma_\lambda \in \mathcal{L}_{j,k}, \tag{4.3}$$

or piecewise bilinear ($d = 2$),

$$\psi_{j,k}(x) = 2^{|\lambda|} \sum_{m=1}^{4} \omega_{j,k,\lambda,m} p_m^{\square}(s), \quad x = \gamma_\lambda(s), \quad s \in \square, \quad \Gamma_\lambda \in \mathcal{L}_{j,k}, \tag{4.4}$$

with certain weights $\omega_{j,k,\lambda} \in \mathbb{R}$ and $\omega_{j,k,\lambda,m} \in \mathbb{R}$, respectively. This *element-based representation* is illustrated in Figure 4.1 in the case of a piecewise bilinear wavelet.

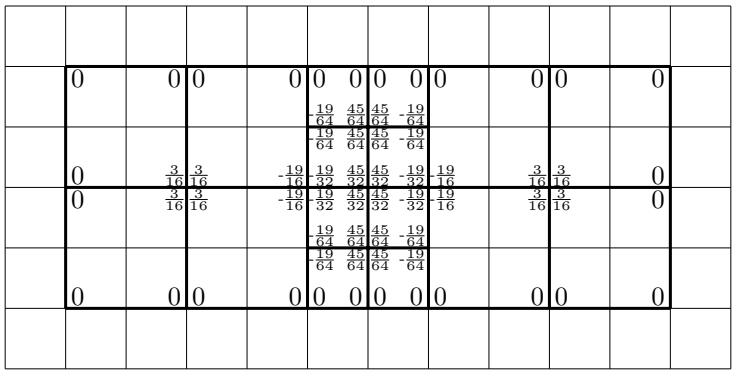

FIG. 4.1. *Element-based representation of a piecewise bilinear wavelet (four vanishing moments).*

Remark. Storing all weights is expensive, particularly for the piecewise bilinears. Since many wavelets correspond to identical masks, a suitable arrangement of the elements in the lists $\mathcal{L}_{j,k}$ induces identical lists of weights for such wavelets. Hence, we suggest storing only the different lists of weights. That way, the required memory is limited to $\mathcal{O}(1)$ since the number of different masks is independent of the level J.

Finally, in view of setting up the compression pattern, we introduce a wavelet tree defined with respect to the supports. Each wavelet has a certain number of sons with

$$\Omega_{j+1,\mathrm{son}} \subseteq \Omega_{j,\mathrm{father}} \qquad \text{if } \psi_{j+1,\mathrm{son}} \text{ is son of } \psi_{j,\mathrm{father}}. \tag{4.5}$$

We mention that the father-son relation might be described by fixed index operations independently of the geometry. However, the tree depends on the chosen wavelet construction.

5. Computing distances numerically. To determine the degrees of quadrature it is necessary to evaluate the distance between elements. Additionally, for the matrix compression, the distance between the supports of two wavelets, as well as the distance between the support of one wavelet and the singular support of another wavelet, has to be computed.

5.1. Computing distances between elements. For each element Γ_λ we determine a sphere $B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda}) := \{x \in \mathbb{R}^3 : \|x - m_{\Gamma_\lambda}\| \le r_{\Gamma_\lambda}\}$ which encloses the element. Then, the distance between elements is approximated by

$$\begin{aligned} \operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'}) &\ge \operatorname{dist}\big(B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda}), B(m_{\Gamma_{\lambda'}}, r_{\Gamma_{\lambda'}})\big) \\ &= \max\{0, \|m_{\Gamma_\lambda} - m_{\Gamma_{\lambda'}}\| - r_{\Gamma_\lambda} - r_{\Gamma_{\lambda'}}\}. \end{aligned} \tag{5.1}$$

We start on the finest level $J = |\lambda|$ and compute the smallest sphere $B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda})$ containing the four vertices of a given element Γ_λ. Assuming that J is sufficiently large, this sphere satisfies $\Gamma_\lambda \subseteq B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda})$. Next, due to

$$\Gamma_\lambda = \bigcup_{\Gamma_{\lambda'} \text{ is son of } \Gamma_\lambda} \Gamma_{\lambda'},$$

the spheres to the elements on the lower levels are obtained recursively by determining the smallest spheres that contain the spheres of the associated sons. This ensures $\Gamma_\lambda \in B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda})$ for all $|\lambda| \le J$ even if the surface Γ is strongly curved.

In the next subsection, the computation of the distance of the singular support and the support of wavelets is reduced to the computation of the distance $\operatorname{dist}(\Gamma_{i,j,k}, \partial\Gamma_{i',j',k'})$ $(j > j')$. Clearly, we find

$$\operatorname{dist}(\Gamma_{i,j,k}, \partial\Gamma_{i',j',k'}) = \operatorname{dist}(\Gamma_{i,j,k}, \Gamma_{i',j',k'}), \qquad i \ne i', \tag{5.2}$$

which is evaluated according to (5.1). Thus, it remains to consider $i = i'$. Since γ_i defines a diffeomorphism, we deduce $\operatorname{dist}(\Gamma_{i,j,k}, \partial\Gamma_{i,j',k'}) \sim \operatorname{dist}(\square_{j,k}, \partial\square_{j',k'})$; i.e. we might reduce our problem to the unit square. Observing that the ℓ^2-norm and the ℓ^∞-norm are equivalent in $\mathbb{R}^2$, the square $\square_{j,k}$ might be interpreted as the sphere $B(m_{\square_{j,k}}, r_{\square_{j,k}})$ with the midpoint $m_{\square_{j,k}} := \gamma_i\big(2^{-(j+1)}\left[\begin{smallmatrix} 2k_1+1 \\ 2k_2+1 \end{smallmatrix}\right]\big)$ and the radius $r_{\square_{j,k}} := 2^{-(j+1)}$. Therefore, we conclude

$$\operatorname{dist}(\Gamma_{i,j,k}, \partial\Gamma_{i,j',k'}) \sim \big|\|m_{\square_{j,k}} - m_{\square_{j',k'}}\|_\infty - r_{\square_{j',k'}}\big| - r_{\square_{j,k}} \tag{5.3}$$

since either $\square_{j,k} \subseteq \square_{j',k'}$ or $\square^\circ_{j,k} \cap \square^\circ_{j',k'} = \emptyset$.

5.2. Computing distances between wavelets. Computing for each wavelet the sphere $B(m_{\psi_{j,k}}, r_{\psi_{j,k}})$ with

$$B(m_{\psi_{j,k}}, r_{\psi_{j,k}}) := \inf_{B(m,r)\in\mathbb{R}^3} \left\{ B(m,r) \supseteq \bigcup_{\Gamma_\lambda \in \mathcal{L}_{j,k}} B(m_{\Gamma_\lambda}, r_{\Gamma_\lambda}) \right\}, \tag{5.4}$$

we conclude from (4.2) that $\Omega_{j,k} \subseteq B(m_{\psi_{j,k}}, r_{\psi_{j,k}})$. Thus, the distance between the wavelets $\psi_{j,k}$ and $\psi_{j',k'}$ is computed analogously to (5.1) by

$$\operatorname{dist}(\Omega_{j,k}, \Omega_{j',k'}) \geq \max\{0, \|m_{\psi_{j,k}} - m_{\psi_{j',k'}}\| - r_{\psi_{j,k}} - r_{\psi_{j',k'}}\}. \tag{5.5}$$

Remark. Of course, it suffices to compute only appropriate approximations to the infimum in (5.4). Moreover, one can use also axis-parallel boxes instead of spheres, which offers the advantage of resolving the high aspect ratios of the wavelets. However, one obtains different distances when rotating the geometry.

Next, the computation of $\operatorname{dist}(\Omega_{j,k}, \Omega'_{j',k'})$ is performed by invoking $\mathcal{L}_{j,k}$ and $\mathcal{L}_{j',k'}$. One readily infers

$$\operatorname{dist}(\Omega_{j,k}, \Omega'_{j',k'}) \gtrsim \min_{\Gamma_\lambda \in \mathcal{L}_{j,k}} \min_{\Gamma_{\lambda'} \in \mathcal{L}_{j',k'}} \big\{ \operatorname{dist}(\Gamma_\lambda, \partial\Gamma_{\lambda'}) \big\},$$

which is evaluated by (5.2) and (5.3), respectively.

6. Setting up the compression pattern. Checking the distance criteria (3.2) for each matrix coefficient in order to assemble the compressed matrix would require $\mathcal{O}(N_J^2)$ function calls. To realize linear complexity, we exploit the tree structure with respect to the supports of the wavelets to predict negligible matrix coefficients. Recall that each son $\psi_{j+1,\text{son}}$ of the wavelet $\psi_{j,\text{father}}$ satisfies $\Omega_{j+1,\text{son}} \subseteq \Omega_{j,\text{father}}$. The following observation is an immediate consequence of the relations $\mathcal{B}_{j,j'} \geq \mathcal{B}_{j+1,j'} \geq \mathcal{B}_{j+1,j+1'}$ and $\mathcal{B}'_{j,j'} \geq \mathcal{B}'_{j+1,j'}$ for $j > j'$.

LEMMA 6.1. *For $\Omega_{j+1,\text{son}} \subseteq \Omega_{j,\text{father}}$ and $\Omega_{j'+1,\text{son}'} \subseteq \Omega_{j',\text{father}'}$ the following statements hold.*

1. $\operatorname{dist}(\Omega_{j,\text{father}}, \Omega_{j',\text{father}'}) > \mathcal{B}_{j,j'}$ *implies* $\operatorname{dist}(\Omega_{j+1,\text{son}}, \Omega_{j',\text{father}'}) > \mathcal{B}_{j+1,j'}$ *as well as* $\operatorname{dist}(\Omega_{j+1,\text{son}}, \Omega_{j'+1,\text{son}'}) > \mathcal{B}_{j+1,j+1'}$.
2. *Suppose that $j > j'$ and* $\operatorname{dist}(\Omega_{j,\text{father}}, \Omega'_{j',\text{father}'}) > \mathcal{B}'_{j,j'}$. *Then one has* $\operatorname{dist}(\Omega_{j+1,\text{son}}, \Omega'_{j',\text{father}'}) > \mathcal{B}'_{j+1,j'}$.

Thanks to this lemma we have to check the distance criteria only for coefficients that stem from subdivisions of calculated coefficients on a coarser level. Obviously, the resulting procedure of checking the distance criteria is then of linear complexity.

7. Assembling the system matrix. This section is concerned with assembling the relevant matrix coefficients $\langle A\psi_{j',k'}, \psi_{j,k}\rangle$. For applying product Gauss quadrature rules it is necessary to split the wavelets into the smooth parts of their supports.

First, for piecewise constant wavelets we introduce the *element-element interactions* by

$$\alpha_{\lambda,\lambda'} := 2^{|\lambda|+|\lambda'|} \int_\square \int_\square k\big(\gamma_\lambda(s), \gamma_{\lambda'}(t)\big)\kappa_\lambda(s)\kappa_{\lambda'}(t)dtds. \tag{7.1}$$

Then, from the wavelet representation (4.3) we conclude

$$\langle A\psi_{j',k'}, \psi_{j,k}\rangle = \sum_{\Gamma_\lambda \in \mathcal{L}_{j,k}} \sum_{\Gamma_{\lambda'} \in \mathcal{L}_{j',k'}} \omega_{j,k,\lambda}\omega_{j',k',\lambda'}\alpha_{\lambda,\lambda'}. \tag{7.2}$$

For piecewise bilinear wavelets, the element-element interactions are given by

$$\beta_{(m,\lambda),(m',\lambda')} := 2^{|\lambda|+|\lambda'|} \int_{\square}\int_{\square} k\big(\gamma_\lambda(s),\gamma_{\lambda'}(t)\big) p_m^{\square}(s) p_{m'}^{\square}(t)\kappa_\lambda(s)\kappa_{\lambda'}(t)dtds. \tag{7.3}$$

Thus, the representation (4.4) yields the equation

$$\langle A\psi_{j',k'},\psi_{j,k}\rangle = \sum_{\Gamma_\lambda\in\mathcal{L}_{j,k}}\sum_{\Gamma_{\lambda'}\in\mathcal{L}_{j',k'}}\sum_{m,m'=1}^{4} \omega_{j,k,\lambda,m}\omega_{j',k',\lambda',m'}\beta_{(m,\lambda),(m',\lambda')}. \tag{7.4}$$

Consequently, the computation of the stiffness matrix is reduced to the computation of element-element interactions. Moreover, since the support of a wavelet on the level j is subdivided into finitely many elements of the levels j and $j+1$, it suffices to compute the element-element interactions $\alpha_{\lambda,\lambda'}$ and $\beta_{(m,\lambda),(m',\lambda')}$ with the precision $\varepsilon_{|\lambda|,|\lambda'|}$ according to (3.3).

8. On element-element interactions. As we have seen in the last section, the assembling of the compressed system matrix reduces to the computation of element-element interactions. Utilizing the mapping θ from (4.1), these element-element interactions are identified uniquely with coefficients of a (sparse) matrix $\mathbf{Q}_J$, where, in the case of piecewise bilinear wavelets, a single entry consists of 16 double values. The pattern of $\mathbf{Q}_J$ is structured similarly to the corresponding compressed system matrix; namely, it is symmetric and *finger structured*. In particular, the symmetry of the pattern suggests computing the element-element interaction of Γ_λ with $\Gamma_{\lambda'}$ and that of $\Gamma_{\lambda'}$ with Γ_λ simultaneously since the application of our quadrature algorithm yields many identical function calls.

Nearly all element-element interactions of $\mathbf{Q}_J$ represent singular or nearly singular integrals. Hence, we often have to subdivide the associated domain of integration for quadrature. We realize this subdivision efficiently by the following *recycling formulas* yielding again element-element interactions which can be *recycled* when assembling the compressed system matrix. Considering a given element Γ_{father} with index father $= (i,j,k)$, the indices of its sons are given by

$$\begin{aligned} &\text{son}_1 := (i,j+1,(2k_1,2k_2)), && \text{son}_2 := (i,j+1,(2k_1+1,2k_2)),\\ &\text{son}_3 := (i,j+1,(2k_1,2k_2+1)), && \text{son}_4 := (i,j+1,(2k_1+1,2k_2+1)). \end{aligned}$$

In the case of piecewise constants, by (7.1) one readily verifies the recycling formulas

$$\begin{aligned} \alpha_{\text{father},\lambda} &= (\alpha_{\text{son}_1,\lambda}+\alpha_{\text{son}_2,\lambda}+\alpha_{\text{son}_3,\lambda}+\alpha_{\text{son}_4,\lambda})/2,\\ \alpha_{\lambda,\text{father}} &= (\alpha_{\lambda,\text{son}_1}+\alpha_{\lambda,\text{son}_2}+\alpha_{\lambda,\text{son}_3}+\alpha_{\lambda,\text{son}_4})/2. \end{aligned} \tag{8.1}$$

Similar formulas can be derived in the case of piecewise bilinears; for example, we have

$$\begin{aligned} \beta_{(1,\text{father}),(m,\lambda)} &= \beta_{(1,\text{son}_1),(m,\lambda)}/2\\ &\quad + (\beta_{(2,\text{son}_1),(m,\lambda)}+\beta_{(4,\text{son}_1),(m,\lambda)}+\beta_{(1,\text{son}_2),(m,\lambda)}+\beta_{(1,\text{son}_3),(m,\lambda)})/4\\ &\quad + (\beta_{(3,\text{son}_1),(m,\lambda)}+\beta_{(4,\text{son}_2),(m,\lambda)}+\beta_{(2,\text{son}_3),(m,\lambda)}+\beta_{(1,\text{son}_4),(m,\lambda)})/8,\\ \beta_{(m,\lambda),(1,\text{father})} &= \beta_{(m,\lambda),(1,\text{son}_1)}/2\\ &\quad + (\beta_{(m,\lambda),(2,\text{son}_1)}+\beta_{(m,\lambda),(4,\text{son}_1)}+\beta_{(m,\lambda),(1,\text{son}_2)}+\beta_{(m,\lambda),(1,\text{son}_3)})/4\\ &\quad + (\beta_{(m,\lambda),(3,\text{son}_1)}+\beta_{(m,\lambda),(4,\text{son}_2)}+\beta_{(m,\lambda),(2,\text{son}_3)}+\beta_{(m,\lambda),(1,\text{son}_4)})/8. \end{aligned} \tag{8.2}$$

Since a finite number of element-element interactions has to be calculated per relevant matrix coefficient, the number of nonzero coefficients of $\mathbf{Q}_J$ is $\mathcal{O}(N_J)$. However, we recommend not storing $\mathbf{Q}_J$ completely since this is so expensive that the computable number of unknowns is crucially restricted. Instead, we propose an element-based assembling of the system matrix. For a fixed element Γ_λ one computes and stores all relevant element-element interactions $\alpha_{\lambda,\lambda'}$ and $\alpha_{\lambda',\lambda}$ with $|\lambda| \geq |\lambda'|$ by the quadrature algorithm introduced in the next section, and likewise for the piecewise bilinears. Then, after updating the system matrix, these values can be deleted. According to $\varepsilon_{j,j'} \leq \varepsilon_{j+1,j'} \leq \varepsilon_{j+1,j'+1}$ for all $j_0 - 1 \leq j \leq j' < J$ (cf. (3.6)), the element-element interactions just have to be computed successively by starting on the coarse grid. Clearly, the proposed strategy requires a rearrangement of the loops within the wavelet Galerkin scheme. However, the number of multiple calculations is reduced enormously, while the requirement of memory is not too high.

9. Numerical integration. As we have seen in the previous sections it suffices to compute element-element interactions with respect to the domain of integration $\square \times \square$. Based on tensor product Gauss–Legendre rules we construct an adaptive quadrature algorithm which converges exponentially. In combination with Theorem 3.2 we realize linear complexity for the computation of the compressed system matrix.

9.1. Error estimates on the reference domain. For a given function $f \in C([0,1])$, we set $I^{[0,1]}f := \int_0^1 f(s)\,ds$. The g-point Gauss–Legendre formula on $[0,1]$ $Q_g^{[0,1]}f := \sum_{i=1}^g \omega_{g,i} f\big(\xi_{g,i}\big)$ applied to $f \in C^{2g}(0,1)$ produces the error

$$\big|R_g^{[0,1]}f\big| := \big|I^{[0,1]}f - Q_g^{[0,1]}f\big| = \frac{2^{4n+1}(n!)^4}{[(2n)!]^3(2n+1)}\big|f^{(2g)}(\xi)\big|, \qquad 0 < \xi < 1;$$

cf. [19], for example. Employing Stirling's formula $n! \sim \sqrt{2\pi n}n^n e^{-n}$, we arrive at the estimate

$$\big|R_g^{[0,1]}f\big| \lesssim \frac{2^{-4g}}{(2g)!} \max_{s\in[0,1]} \big|f^{(2g)}(s)\big|. \tag{9.1}$$

For $f \in C(\square)$ we define $I^\square f := \big(I^{[0,1]} \otimes I^{[0,1]}\big)f = \int_\square f(s)\,ds$ with $s = \left[\begin{smallmatrix} s_1 \\ s_2 \end{smallmatrix}\right]$. Approximating $I^\square f$ by the product Gauss–Legendre quadrature formula

$$Q_g^\square f := \big(Q_g^{[0,1]} \otimes Q_g^{[0,1]}\big)f = \sum_{i,i'=1}^g \omega_{g,i}\omega_{g,i'} f\big(\left[\begin{smallmatrix} \xi_{g,i} \\ \xi_{g,i'} \end{smallmatrix}\right]\big), \tag{9.2}$$

we find the following error estimate.

LEMMA 9.1. *If $f \in C^{2g}(\square)$, the quadrature error $R_g^\square f := I^\square f - Q_g^\square f$ of the product Gauss–Legendre quadrature formula* (9.2) *is bounded by*

$$\big|R_g^\square f\big| \lesssim \frac{2^{-4g}}{(2g)!}\Big[\max_{s\in\square}\big|\partial_{s_1}^{2g} f(s)\big| + \max_{s\in\square}\big|\partial_{s_2}^{2g} f(s)\big|\Big]. \tag{9.3}$$

Proof. Invoking (9.1), the classical tensor product argument

$$\begin{aligned} R_g^\square f &= \Big[\big(I^{[0,1]} \otimes I^{[0,1]}\big) - \big(Q_g^{[0,1]} \otimes Q_g^{[0,1]}\big)\Big]f \\ &= \Big[\big(I^{[0,1]} \otimes I^{[0,1]}\big) - \big(I^{[0,1]} \otimes Q_g^{[0,1]}\big) + \big(I^{[0,1]} \otimes Q_g^{[0,1]}\big) - \big(Q_g^{[0,1]} \otimes Q_g^{[0,1]}\big)\Big]f \\ &= \Big[I^{[0,1]} \otimes \big(I^{[0,1]} - Q_g^{[0,1]}\big)\Big]f + \Big[\big(I^{[0,1]} - Q_g^{[0,1]}\big) \otimes Q_g^{[0,1]}\Big]f \end{aligned}$$

leads us to the desired estimate (9.3). ∎

Next, to a function $f(s,t) \in C(\square \times \square)$ we apply the four-dimensional product Gauss–Legendre quadrature formula

$$Q_{g,g'}^{\square\times\square} f := \big(Q_g^{\square} \otimes Q_{g'}^{\square}\big) f = \sum_{i,i'=1}^{g} \sum_{j,j'=1}^{g'} \omega_{g,i}\omega_{g,i'}\omega_{g',j}\omega_{g',j'} f\Big(\Big[\begin{smallmatrix}\xi_{g,i}\\ \xi_{g,i'}\end{smallmatrix}\Big], \Big[\begin{smallmatrix}\xi_{g',j}\\ \xi_{g',j'}\end{smallmatrix}\Big]\Big) \tag{9.4}$$

in order to approximate the integral $I^{\square\times\square} f := \big(I^{\square} \otimes I^{\square}\big) f = \int_{\square}\int_{\square} f(s,t)\, dt\, ds$.

LEMMA 9.2. *For $f(s,t) \in C^{2g}(\square) \times C^{2g'}(\square)$ the quadrature error $R_{g,g'}^{\square\times\square} f := I^{\square\times\square} f - Q_{g,g'}^{\square\times\square} f$ of the product Gauss–Legendre quadrature formula* (9.4) *can be estimated by*

$$\begin{aligned}\big|R_{g,g'}^{\square\times\square} f\big| \lesssim{}& \frac{2^{-4g}}{(2g)!}\Big[\max_{s,t\in\square}\big|\partial_{s_1}^{2g} f(s,t)\big| + \max_{s,t\in\square}\big|\partial_{s_2}^{2g} f(s,t)\big|\Big] \\ &+ \frac{2^{-4g'}}{(2g')!}\Big[\max_{s,t\in\square}\big|\partial_{t_1}^{2g'} f(s,t)\big| + \max_{s,t\in\square}\big|\partial_{t_2}^{2g'} f(s,t)\big|\Big].\end{aligned}$$

Proof. This lemma is proven analogously to the previous lemma. ∎

9.2. Basic estimates. We now assume that the diffeomorphisms γ_i are analytical on $\square$ for all $i \in \{1,2,\dots,M\}$. Then, in general, a given boundary integral operator $A : H^q(\Gamma) \to H^{-q}(\Gamma)$ of order $2q$ satisfies the following definition.

DEFINITION 9.3. *A kernel $k(x,y)$ is called analytically standard of order $2q$ if the partial derivatives of the transported kernel functions* (1.6) *are bounded by*

$$|\partial_s^{\alpha}\partial_t^{\beta} k_{i,i'}(s,t)| \lesssim \frac{(|\alpha|+|\beta|)!}{r^{|\alpha|+|\beta|}} \big\|\gamma_i(s) - \gamma_{i'}(t)\big\|^{-(2+2q+|\alpha|+|\beta|)}$$

with some $r > 0$, provided that $2 + 2q + |\alpha| + |\beta| > 0$.

Defining the local transported kernels

$$k_{\lambda,\lambda'}(s,t) := k\big(\gamma_\lambda(s), \gamma_{\lambda'}(t)\big)\kappa_\lambda(s)\kappa_{\lambda'}(t) dt ds$$

for $s,t \in \square$, we find the relation

$$I^{\square\times\square} k_{\lambda,\lambda'} = 2^{-(|\lambda|+|\lambda'|)} \alpha_{\lambda,\lambda'}. \tag{9.5}$$

LEMMA 9.4. *Let the kernel $k(x,y)$ be analytically standard of order $2q$ and assume that $\operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'}) > 0$. Then, applying the quadrature formula $Q_{g,g'}^{\square\times\square}$ defined in* (9.4) *to the integral* (9.5) *yields the error estimate*

$$\begin{aligned}\big|R_{g,g'}^{\square\times\square} k_{\lambda,\lambda'}\big| \lesssim 2^{-2(|\lambda|+|\lambda|')}\Bigg[&\left(\frac{2^{-|\lambda|}}{4r}\right)^{2g} \operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'})^{-(2+2q+2g)} \\ &+ \left(\frac{2^{-|\lambda'|}}{4r}\right)^{2g'} \operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'})^{-(2+2q+2g')}\Bigg].\end{aligned} \tag{9.6}$$

Proof. Since the kernel is analytically standard of order $2q$, (1.4) and (1.5) imply the estimate

$$|\partial_s^{\alpha}\partial_t^{\beta} k_{\lambda,\lambda'}(s,t)| \lesssim \frac{(|\alpha|+|\beta|)!}{r^{|\alpha|+|\beta|}} \frac{2^{-|\lambda|(|\alpha|+2)} 2^{-|\lambda'|(|\beta|+2)}}{\big\|\gamma_\lambda(s) - \gamma_{\lambda'}(t)\big\|^{2+2q+|\alpha|+|\beta|}}. \tag{9.7}$$

Hence, we find

$$\max_{s,t\in\square}\left|\partial_{s_n}^{2g}k_{\lambda,\lambda'}(s,t)\right| \lesssim 2^{-2(|\lambda|+|\lambda'|)}\ 2^{-2g|\lambda|}\ \frac{(2g)!}{r^{2g}}\ \min_{s,t\in\square}\left\|\gamma_\lambda(s)-\gamma_{\lambda'}(t)\right\|^{-(2+2q+2g)}$$
$$\lesssim 2^{-2(|\lambda|+|\lambda'|)}\ 2^{-2g|\lambda|}\ \frac{(2g)!}{r^{2g}}\ \operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{-(2+2q+2g)}$$

for $n=1,2$. Analogously one infers

$$\max_{s,t\in\square}\left|\partial_{t_n}^{2g'}k_{\lambda,\lambda'}(s,t)\right| \lesssim 2^{-2(|\lambda|+|\lambda'|)}\ 2^{-2g'|\lambda'|}\ \frac{(2g')!}{r^{2g'}}\ \operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{-(2+2q+2g')}.$$

The integrand $k_{\lambda,\lambda'}$ is nonsingular if $\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})>0$. Therefore, we conclude the desired estimate by Lemma 9.2. ∎

In the case of piecewise bilinear wavelets, we have to compute in accordance with (7.3) the integrals

$$I^{\square\times\square}\big[k_{\lambda,\lambda'}(p_m^\square\otimes p_{m'}^\square)\big]=2^{-(|\lambda|+|\lambda'|)}\beta_{(m,\lambda),(m',\lambda')}. \tag{9.8}$$

Herein, $p_m^\square$, $p_{m'}^\square$ denote the bilinear polynomials on $\square$ defined by (2.2).

LEMMA 9.5. *Let the kernel $k(x,y)$ be analytically standard of order $2q$ and assume that $\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})\gtrsim 2^{-\min\{|\lambda|,|\lambda'|\}}$. Then, the application of the quadrature formula $Q_{g,g'}^{\square\times\square}$ (9.4) to the integral (9.8) yields the error estimate*

$$\left|R_{g,g'}^{\square\times\square}\big[k_{\lambda,\lambda'}(p_m^\square\otimes p_{m'}^\square)\big]\right| \lesssim \left(\frac{2^{-|\lambda|}}{4r}\right)^{2g}\frac{2^{-2|\lambda|'-|\lambda|}}{\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{1+2q+2g}} + \left(\frac{2^{-|\lambda'|}}{4r}\right)^{2g'}\frac{2^{-2|\lambda|-|\lambda|'}}{\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{1+2q+2g'}}. \tag{9.9}$$

Proof. The $(2g)$th partial derivative of $k_{\lambda,\lambda'}(s,t)p_m^\square(s)p_{m'}^\square(t)$ with respect to s_n, $n=1,2$, is given by

$$\partial_{s_n}^{2g}\big[k_{\lambda,\lambda'}(s,t)p_m^\square(s)p_{m'}^\square(t)\big]=\partial_{s_n}^{2g}k_{\lambda,\lambda'}(s,t)p_m^\square(s)p_{m'}^\square(t) + \partial_{s_n}^{2g-1}k_{\lambda,\lambda'}(s,t)\partial_{s_n}p_m^\square(s)p_{m'}^\square(t).$$

Obviously, there holds $\left|p_m^\square(s)p_{m'}^\square(t)\right|\le 1$ and $\left|\frac{\partial p_m^\square(s)}{\partial s_n}p_{m'}^\square(t)\right|\le 1$ for all $s,t\in\square$ and $1\le m,m'\le 4$; cf. (2.2). Therefore, according to (9.7) we find the bound

$$\max_{s,t\in\square}\left|\partial_{s_n}^{2g}\big[k_{\lambda,\lambda'}(s,t)p_m^\square(s)p_{m'}^\square(t)\big]\right| \lesssim 2^{-2g|\lambda|}\frac{(2g)!}{r^{2g}}\frac{2^{-2|\lambda'|-|\lambda|}}{\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{1+2q+2g}}\cdot\underbrace{\left[\frac{2^{-|\lambda|}}{\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})}+\frac{r}{2g}\right]}_{\lesssim 1\text{ since }\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})\gtrsim 2^{-\min\{|\lambda|,|\lambda'|\}}}$$
$$\lesssim 2^{-2g|\lambda|}\frac{(2g)!}{r^{2g}}\frac{2^{-2|\lambda'|-|\lambda|}}{\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'})^{1+2q+2g}}.$$

Analogously one gets the corresponding result for the derivatives with respect to t_1 and t_2. Hence, (9.9) is valid according to Lemma 9.2. ∎

We now can formulate the following proposition, which is an immediate consequence of Lemmas 9.4 and 9.5.

PROPOSITION 9.6. *Let the kernel $k(x,y)$ be analytically standard of order $2q$ and let ε denote a given precision. Consider two elements Γ_λ and $\Gamma_{\lambda'}$ which satisfy the distance criterion*

$$\operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'}) \geq 2^{-\min\{|\lambda|,|\lambda'|\}} s, \qquad s > \frac{1}{4r}. \tag{9.10}$$

Then, the Gauss–Legendre quadrature formula $Q_{g,g'}^{\square\times\square}$ (9.4) computes the element-element interactions $\alpha_{\lambda,\lambda'}$ and $\beta_{(m,\lambda),(m',\lambda')}$, respectively, with precision $\lesssim \varepsilon$ if we choose the degrees of quadrature according to

$$\begin{aligned} g &= \left\lceil -\tfrac{1}{2} \cdot \frac{|\lambda|+|\lambda'|+\log_2(\varepsilon)+(2+2q)\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))}{|\lambda|+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))} \right\rceil, \\ g' &= \left\lceil -\tfrac{1}{2} \cdot \frac{|\lambda|+|\lambda'|+\log_2(\varepsilon)+(2+2q)\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))}{|\lambda'|+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))} \right\rceil \end{aligned} \tag{9.11}$$

in the case of piecewise constant wavelets and

$$\begin{aligned} g &= \left\lceil -\tfrac{1}{2} \cdot \frac{|\lambda'|+\log_2 \varepsilon+(1+2q)\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))}{|\lambda|+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))} \right\rceil, \\ g' &= \left\lceil -\tfrac{1}{2} \cdot \frac{|\lambda|+\log_2 \varepsilon+(1+2q)\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))}{|\lambda'|+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_\lambda,\Gamma_{\lambda'}))} \right\rceil \end{aligned} \tag{9.12}$$

in the case of piecewise bilinear wavelets.

Notice that the error of quadrature does not tend to zero when increasing g and g' if the elements violate the distance criterion (9.10). For such integrals we propose an adaptive quadrature strategy in the next subsection.

A special situation occurs if $|\lambda| = |\lambda'|$ and $\Gamma_\lambda \cap \Gamma_{\lambda'} \neq \emptyset$, i.e., if both elements are identical or share a common edge or vertex. Then, the domain of integration contains the singularity. We utilize the *Duffy trick* to transform the singular integrands on nonsingular ones; cf. [15, 29, 30]. It suffices to consider an error of quadrature which behaves as in (9.6) and (9.9), respectively, with $\operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'}) := 2^{-|\lambda|}$. Consequently, we have to subdivide the domain of integration $[0,1]^4$ into 16^M equal sized cubes. Herein, M denotes the nonnegative integer $M := \max\{0, \lceil \log_2 s \rceil\}$ with s from (9.10). After rescaling the domains of integration to $[0,1]^4$, the application of the quadrature formula $Q_{g,g'}^{\square\times\square}$ with

$$g = g' = \left\lceil \frac{q|\lambda| - (\log_2 \varepsilon)/2}{2+M+\log_2 r} \right\rceil \tag{9.13}$$

yields a quadrature error $\lesssim \varepsilon$.

Remark. A refined analysis shows that the choice (9.13) can be weakened. Indeed, in certain coordinates the degree of quadrature can be chosen smaller; see [30] for the details.

9.3. An adaptive quadrature strategy. We are now in the position to formulate the recursive adaptive algorithm. We prove in Lemma 9.8 that the algorithm computes the element-element interaction of Γ_λ and $\Gamma_{\lambda'}$ with the precision $\varepsilon_{|\lambda|,|\lambda'|}$. Without loss of generality we assume $|\lambda'| \geq |\lambda|$. An example of the subdivision of the elements is depicted in Figure 9.1.

1. *Starting point.* If the elements Γ_λ and $\Gamma_{\lambda'}$ violate the distance criterion (9.10), then goto item 2 if $|\lambda'| > |\lambda|$ or goto item 3 if $|\lambda| = |\lambda'|$. Otherwise apply the product Gauss–Legendre quadrature formula $Q_{g,g'}^{\square\times\square}$ choosing g and g' associated with the precision $\varepsilon_{|\lambda|,|\lambda'|}$ as in Proposition 9.6.

2. *Case* $|\lambda'| > |\lambda|$. Replace the larger element Γ_λ by its four sons and compute the associated element-element interactions with precision $\varepsilon_{|\lambda|,|\lambda'|}$ according to item 1. The desired element-element interaction is calculated via the recycling formulas.
3. *Case* $|\lambda| = |\lambda'|$. If the domain of integration contains the singularity, then apply the Duffy trick. Else, replace both elements Γ_λ and $\Gamma_{\lambda'}$ by their sons and compute the associated element-element interactions with precision $\varepsilon_{|\lambda|,|\lambda'|}$ according to item 1. The desired element-element interaction is calculated via the recycling formulas.

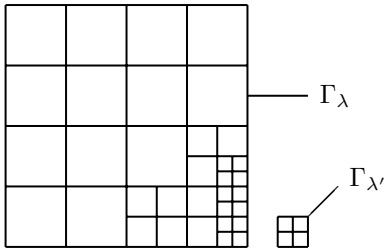

FIG. 9.1. *This subdivision of the elements* Γ_λ *and* $\Gamma_{\lambda'}$ *results if* $s = 3/2$.

Notice that the recycling is easily realized by inserting "return the previously computed element-element interaction if it exists" in item 1. The next lemma shows that the algorithm stops after at most $\mathcal{O}(||\lambda| - |\lambda'||)$ subdivision steps.

LEMMA 9.7. *The following statements concerning the computation of the element-element interaction* $\alpha_{\lambda,\lambda'}$ *or* $\beta_{(m,\lambda),(m',\lambda')}$ *by the above quadrature algorithm are valid:*

1. *The given element-element interaction is subdivided into at most* $\mathcal{O}(||\lambda|-|\lambda'||)$ *element-element interactions* $\alpha_{\widehat{\lambda},\widehat{\lambda}'}$ *and* $\beta_{(m,\widehat{\lambda}),(m',\widehat{\lambda}')}$, *where* $|\widehat{\lambda}| \geq |\lambda|$, $|\widehat{\lambda}'| \geq |\lambda'|$.
2. *If* $|\lambda| \leq |\lambda'|$, *there holds* $|\lambda| \leq |\widehat{\lambda}| \leq |\widehat{\lambda}'| \sim |\lambda'|$. *The analogous result holds if* $|\lambda'| \leq |\lambda|$.
3. *On fixed levels* $|\widehat{\lambda}|$ *and* $|\widehat{\lambda}'|$, *the number of both directly computed as well as subdivided element-element interactions* $\alpha_{\widehat{\lambda},\widehat{\lambda}'}$ *and* $\beta_{(m,\widehat{\lambda}),(m',\widehat{\lambda}')}$ *is* $\mathcal{O}(1)$.

Proof. There is nothing to prove in the case of two elements Γ_λ and $\Gamma_{\lambda'}$ satisfying the distance criterion (9.10) or if the Duffy trick is applied. Hence, we consider first $|\lambda| = |\lambda'|$ and two disjoint elements Γ_λ and $\Gamma_{\lambda'}$ which violate the distance criterion (9.10). Since the mesh is quasi-uniform, a constant $c_\Gamma > 0$ exists such that $\operatorname{dist}(\Gamma_\lambda, \Gamma_{\lambda'}) \geq 2^{-|\lambda|} c_\Gamma$ for all pairs of elements with $\Gamma_\lambda \cap \Gamma_{\lambda'} = \emptyset$. Herein, c_Γ depends on only the manifold and its parametrization but not on $|\lambda|$ or J. The algorithm subdivides the desired element-element interaction into at most 16^M element-element interactions $\alpha_{\widehat{\lambda},\widehat{\lambda}'}$ or $\beta_{(m,\widehat{\lambda}),(m',\widehat{\lambda}')}$, where $|\lambda| \leq |\widehat{\lambda}| = |\widehat{\lambda}'| \leq |\lambda| + M$ and $M = \max\{0, \lceil \log_2(s/c_\Gamma) \rceil\}$.

Finally, we consider the case of two elements Γ_λ and $\Gamma_{\lambda'}$, $|\lambda| \neq |\lambda'|$, violating the distance criterion (9.10). Without loss of generality we assume $|\lambda| < |\lambda'|$. Moreover, we make use of the following observation. Due to $\operatorname{diam}(\Gamma_{\widehat{\lambda}}) \sim 2^{-|\widehat{\lambda}|}$, for a fixed level $|\widehat{\lambda}| \leq |\lambda'|$, only $\mathcal{O}(s^2)$ elements $\Gamma_{\widehat{\lambda}}$ exist with $\operatorname{dist}(\Gamma_{\widehat{\lambda}}, \Gamma_{\lambda'}) < s 2^{-|\widehat{\lambda}|}$. Consequently, our algorithm subdivides for all $|\lambda| < |\widehat{\lambda}| \leq |\lambda'|$ only $\mathcal{O}(s^2)$ element-element interactions while $\mathcal{O}(s^2)$ element-element interactions are evaluated since they satisfy the

distance criterion. The remaining $\mathcal{O}(s^2)$ element-element interactions associated with $|\widehat{\lambda}| = |\lambda|$ are subdivided into $\mathcal{O}(s^4)$ element-element interactions according to the aforementioned cases. ∎

We prove now that the proposed quadrature algorithm computes the desired element-element interaction with a precision that stays proportional to $\varepsilon_{|\lambda|,|\lambda'|}$.

LEMMA 9.8. *The above quadrature algorithm computes a desired element-element interaction $\alpha_{\lambda,\lambda'}$ or $\beta_{(m,\lambda),(m',\lambda')}$ with a precision that stays proportional to $\varepsilon_{|\lambda|,|\lambda'|}$.*

Proof. Without loss of generality we assume $|\lambda| \le |\lambda'|$. According to Lemma 9.7, the computation of the desired element-element interaction is reduced to the computation of element-element interactions $\alpha_{\widehat{\lambda},\widehat{\lambda}'}$ or $\beta_{(m,\widehat{\lambda}),(m',\widehat{\lambda}')}$, where $|\lambda| \le |\widehat{\lambda}| \le |\widehat{\lambda}'| \sim |\lambda'|$. Hereby, for a fixed pair of levels $|\widehat{\lambda}|, |\widehat{\lambda}'|$, only $\mathcal{O}(1)$ element-element interactions are evaluated. Observing the damping factor $1/2$ in the recycling formulas (8.1) and (8.2), we might estimate the final quadrature error by $\sum_{|\widehat{\lambda}|=|\lambda|}^{|\lambda'|} \varepsilon_{|\lambda|,|\lambda'|} 2^{|\lambda|-|\widehat{\lambda}|} \lesssim \varepsilon_{|\lambda|,|\lambda'|}$. ∎

Consequently, by the above quadrature algorithm, the system matrix is computed consistent with Theorem 3.3. In combination with Theorem 3.2 we prove next that the proposed quadrature strategy in fact computes the compressed system matrix within linear complexity. This is achieved even without the recycling of previously computed element-element interactions. But we like to stress that the recycling accelerates the computation of the system matrix enormously.

THEOREM 9.9. *Exploiting the adaptive quadrature algorithm introduced above, the computation of the element-element interactions $\alpha_{\lambda,\lambda'}$ and $\beta_{(m,\lambda),(m',\lambda')}$ with the precision $\varepsilon_{|\lambda|,|\lambda'|}$ given by* (3.6) *requires $\mathcal{O}\big([2J - |\lambda| - |\lambda'|]^4\big)$ operations.*

Proof. (i) Throughout this proof we will abbreviate $j := |\lambda|$ and $j' := |\lambda'|$. We find $\frac{d'-q}{\tilde{d}+q} < 1$ due to (3.3). Consequently, $\varepsilon_{j,j'}$ can be estimated by

$$\varepsilon_{j,j'} \sim \min\Big\{2^{-|j-j'|}, 2^{-(2J-j-j')\frac{d'-q}{d+q}}\Big\} 2^{2Jq} 2^{-d'(2J-j-j')} \ge 2^{2Jq} 2^{-(d'+1)(2J-j-j')}.$$

Moreover, according to (9.4), the quadrature rule $Q_{g,g'}^{\square\times\square}$ applies $(gg')^2$ quadrature points. In view of Lemma 9.7 we may neglect the evaluation of the recycling formulas since this complexity scales linearly with the number $\mathcal{O}(|j-j'|)$ of computed element-element interactions and $|j - j'| \le 2J - j - j'$.

(ii) We estimate the degrees of quadrature to compute a single element-element interaction $\alpha_{\widehat{\lambda},\widehat{\lambda}'}$ or $\beta_{(m,\widehat{\lambda}),(m,\widehat{\lambda}')}$, which stems from our adaptive quadrature algorithm, with the precision $\varepsilon_{j,j'}$. Without loss of generality we assume $j' \ge j$, which implies $j' \sim \widehat{j}' := |\widehat{\lambda}'| \ge \widehat{j} := |\widehat{\lambda}| \ge j$ according to Lemma 9.7. Quadrature is applied if either the Duffy trick is employed or the distance criterion (9.10) is satisfied.

First, we treat the case that the Duffy trick is applied, which implies $\widehat{j} = \widehat{j}'$. We have to compute 16^M, $M = \max\{0, \lceil \log_2 s \rceil\}$, integrals with the degree of quadrature given by (9.13). The choice of M ensures $2 + M + \log_2 r = c > 0$, which leads to

$$g = g' \sim \tfrac{q\widehat{j} - (\log_2 \varepsilon_{j,j'})/2}{2+M+\log_2 r} \le \tfrac{1}{2c}[-2q(J-\widehat{j}) + (d'+1)(2J-j-j')].$$

Hence, we need $\mathcal{O}\big([2J-j-j']^4\big)$ quadrature points since

$$-2q(J-\widehat{j}) \lesssim \begin{cases} 1 & \text{if } q \ge 0, \\ -2q(2J-j-j') & \text{otherwise} \end{cases} \tag{9.14}$$

and $M \sim 1$ independently of J and $\widehat{j}$.

Next, let the distance criterion (9.10) be satisfied. Thus, there holds

$$\widehat{j} + 2 + \log_2 r + \log_2 \big(\operatorname{dist}(\Gamma_{\widehat{\lambda}}, \Gamma_{\widehat{\lambda}'}) \big) \geq \widehat{j} + 2 + \log_2 r + \log_2(2^{-\widehat{j}} s) =: c > 0.$$

Inserting this inequality into (9.11) yields

$$\begin{aligned} g &\sim -\tfrac{1}{2} \cdot \tfrac{\widehat{j}+\widehat{j}'+\log_2(\varepsilon_{j,j'})+(2+2q)\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))}{\widehat{j}+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))} \\ &\leq -\tfrac{1}{2c}[\widehat{j} + \widehat{j}' + \log_2(\varepsilon_{j,j'}) + (2+2q)\log_2(2^{-\widehat{j}} s)] \\ &= \tfrac{1}{2c}[-2q(J-\widehat{j}) + \widehat{j} - \widehat{j}' + (d'+1)(2J-j-j') + (2+2q)\log_2(1/s)]. \end{aligned}$$

Inequality (9.14) and $\widehat{j} \leq \widehat{j}'$ imply $g \lesssim 2J - j - j'$. In complete analogy we estimate

$$\begin{aligned} g' &\sim -\tfrac{1}{2} \cdot \tfrac{\widehat{j}+\widehat{j}'+\log_2(\varepsilon_{j,j'})+(2+2q)\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))}{\widehat{j}'+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))} \\ &= -\tfrac{1}{2} \cdot \tfrac{\widehat{j}+\widehat{j}'+\log_2(\varepsilon_{j,j'})+(2+2q)\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))}{[\widehat{j}'-\widehat{j}]+[\widehat{j}+2+\log_2 r+\log_2(\operatorname{dist}(\Gamma_{\widehat{\lambda}},\Gamma_{\widehat{\lambda}'}))]} \lesssim \tfrac{1}{\widehat{j}'-\widehat{j}+c}(2J-j-j'). \end{aligned}$$

Consequently, $\mathcal{O}\big(\frac{1}{(\widehat{j}'-\widehat{j}+c_s)^2}[2J-j-j']^4\big)$ quadrature points are evaluated. One readily verifies that both estimates remain valid for the degrees of quadrature given by (9.12), i.e., in the case of piecewise bilinear functions.

(iii) According to Lemma 9.7 we have just to sum over $\widehat{j}$ with $j < \widehat{j} \leq j'$ to estimate the work of computing the desired element-element interaction. From

$$\sum_{\widehat{j}=j+1}^{j'} \frac{1}{(j'-\widehat{j}+c)^2} < \sum_{\widehat{j}=0}^{\infty} \frac{1}{(\widehat{j}+c)^2} < \frac{1}{c^2} + \sum_{\widehat{j}=1}^{\infty} \frac{1}{\widehat{j}^2} = \frac{1}{c^2} + \frac{\pi^2}{6} \lesssim 1$$

we derive indeed the desired complexity $\mathcal{O}\big([2J-j-j']^4\big)$. ∎

10. Numerical results. In order to confirm our analysis and to demonstrate the proposed method we shall present some numerical results in this section. We shall consider the Laplace equation $\Delta u = 0$ in Ω, where we employ first the unit sphere as domain Ω. It is well known that, denoting the spherical harmonics by Y_n^m, the functions of type $u(x) = \|x\|^2 Y_n^m(x/\|x\|)$ are solutions of the Laplace equation in the sphere. We will treat both the Dirichlet problem with data $f := u|_\Gamma = Y_2^0$ and the Neumann problem with data $g := \partial u/\partial n = 2Y_2^0$. Then, $u = \|x\|^2 Y_2^0(x/\|x\|)$ is the unique solution (modulo some constant in the case of the Neumann problem) of the interior Dirichlet and Neumann problem, respectively.

We solve both the interior Dirichlet and the interior Neumann problem by the indirect approach using the *single layer operator*

$$V : H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma), \qquad (V\rho)(x) = \frac{1}{4\pi} \int_\Gamma \frac{1}{\|x-y\|} \rho(y) d\Gamma_y,$$

the *double layer operator* K and its adjoint $K^\star$,

$$K - 1/2 : L^2(\Gamma) \to L^2(\Gamma), \qquad (K\rho)(x) = \frac{1}{4\pi} \int_\Gamma \frac{\partial}{\partial n_y} \frac{1}{\|x-y\|} \rho(y) d\Gamma_y,$$

$$K^\star + 1/2 : L^2(\Gamma)/\mathbb{R} \to L^2(\Gamma)/\mathbb{R}, \qquad (K^\star\rho)(x) = \frac{1}{4\pi} \int_\Gamma \frac{\partial}{\partial n_x} \frac{1}{\|x-y\|} \rho(y) d\Gamma_y,$$

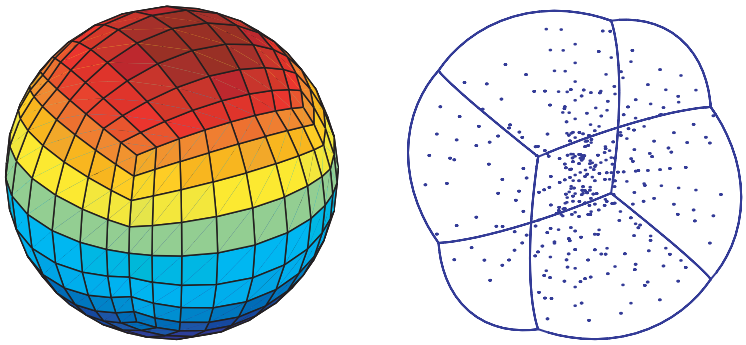

FIG. 10.1. *The mesh on the unit sphere ($j = 3$) and evaluation points of the solution.*

as well as the hypersingular operator

$$W : H^{1/2}(\Gamma)/\mathbb{R} \to H^{-1/2}(\Gamma)/\mathbb{R}, \qquad (W\rho)(x) = -\frac{1}{4\pi}\frac{\partial}{\partial n_x}\int_\Gamma \frac{\partial}{\partial n_y}\frac{1}{\|x-y\|}\rho(y)d\Gamma_y.$$

The indirect method yields a density ρ living on the boundary Γ which leads to the solution $u(x)$ by potential evaluation. More precisely, for the Dirichlet problem we derive the Fredholm integral equations of the first kind,

$$V\rho = f \text{ on } \Gamma, \qquad u(x) = (V\rho)(x) \text{ in } \Omega, \tag{10.1}$$

and of the second kind,

$$\left(K - \frac{1}{2}\right)\rho = f \text{ on } \Gamma, \qquad u(x) = (K\rho)(x) \text{ in } \Omega. \tag{10.2}$$

The Neumann problem is solved via the Fredholm integral equations of the first kind,

$$W\rho = g \text{ on } \Gamma, \qquad u(x) = (K\rho)(x) \text{ in } \Omega, \tag{10.3}$$

and of the second kind,

$$\left(K^\star + \frac{1}{2}\right)\rho = g \text{ on } \Gamma, \qquad u(x) = (V\rho)(x) \text{ in } \Omega. \tag{10.4}$$

The parametric representation of the sphere via six patches is derived by projecting the cube $[-1,1]^3$ onto the unit sphere; cf. Figure 10.1. We shall solve all these integral equations by piecewise constant and bilinear wavelets. Equations (10.1), (10.2), and (10.4) are discretized by piecewise constant wavelets with three vanishing moments as well as piecewise bilinear wavelets with double nodes along the interfaces and four vanishing moments. The discretization of (10.3) requires *globally continuous* piecewise bilinear wavelet bases since the energy space of the hypersingular operator is $H^{1/2}(\Gamma)/\mathbb{R}$. It suffices that the wavelets provide two vanishing moments. Since the kernel is hypersingular, we employ the identity $\langle W\rho, \mu\rangle = \langle V \operatorname{curl}_\Gamma \rho, \operatorname{curl}_\Gamma \mu\rangle$ (cf. [25]).

Since the spherical harmonics are eigenfunctions of all these boundary integral operators, the density ρ is a certain multiple of Y_2^0. Therefore, we are able to measure the $L^2(\Gamma)$-error of the density for which the error estimate $\|\rho - \rho_J\|_{L^2(\Gamma)} \lesssim 2^{-Jd}$

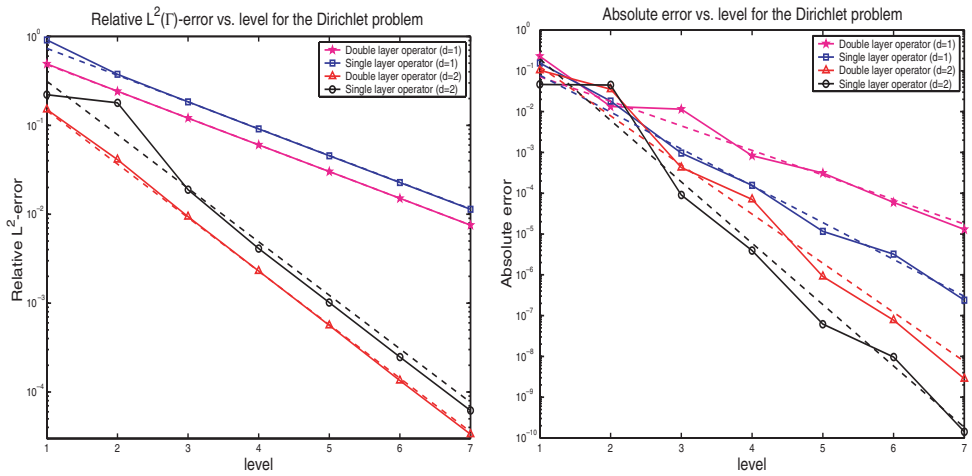

FIG. 10.2. *Dirichlet problem: errors of the density (left) and the potential evaluation (right).*

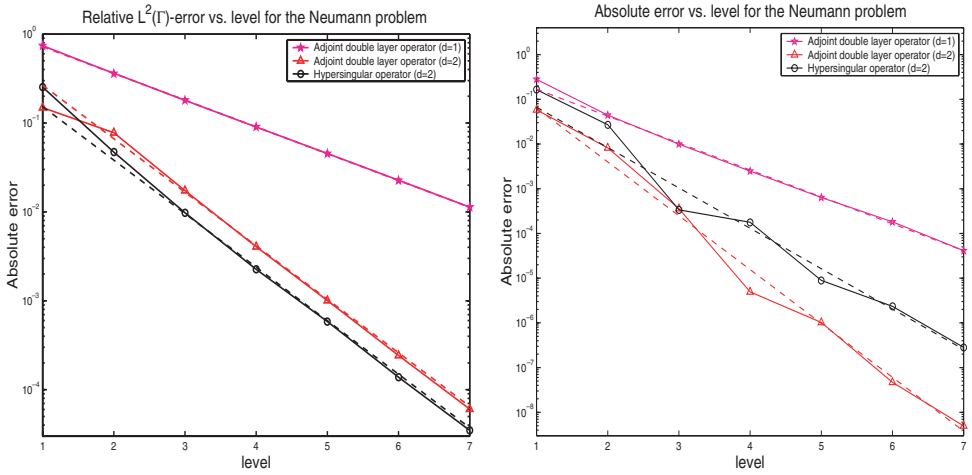

FIG. 10.3. *Neumann problem: errors of the density (left) and the potential evaluation (right).*

holds. In order to measure also the best possible order of convergence, we compute additionally the approximate solution u_J of the Laplace equation in a lot of points x_i distributed in the interior of the sphere as shown in the right plot of Figure 10.1. Depending on the approximation order d and the operator order $2q$, the optimal order of convergence (including the Aubin–Nitsche trick) of $u(x_i) - u_J(x_i)$ in a fixed evaluation point x_i is given by $|u(x_i) - u_J(x_i)| \lesssim 2^{-2J(d-q)}$; cf. [34]. However, the errors in a single point might oscillate extremely due to round-off errors. To be on safe ground we compute the ℓ^∞-norm of the measurements in all points. If this norm converges with the optimal order, we are sure that we achieve the optimal order of convergence. Instead of the ℓ^∞-norm we computed also the mean value and the ℓ^2-norm. But this has not changed the convergence behavior. The convincing results produced by our algorithm are presented in Figures 10.2 and 10.3 for the Dirichlet and the Neumann

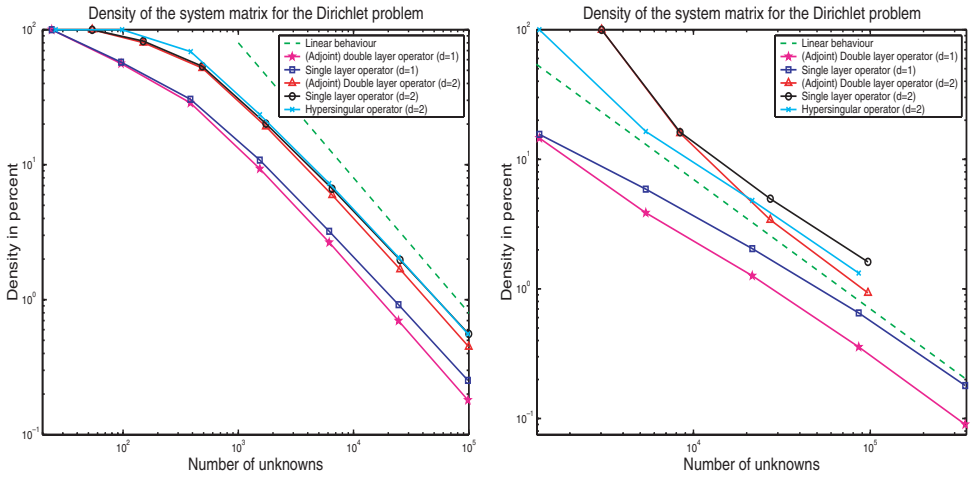

FIG. 10.4. *Compression rates for the sphere (left) and the gearwheel (right).*

problem, respectively. The left-hand plots are concerned with the relative $L^2(\Gamma)$-error of the density, while the right-hand plots show the absolute ℓ^∞-error of the potential evaluation. In all computations, the optimal orders of convergence, indicated by the dashed lines, are realized.

The density of the system matrix, measured by the ratio of relevant coefficients versus N_J^2, is visualized on the left-hand side of Figure 10.4. One figures out that the number of nonzero coefficients tends in fact to a linear behavior (dashed line).

In our second example we consider the gearwheel presented in Figure 1.1 as domain Ω and choose, as above, the Dirichlet or Neumann data of the harmonical function $u(x) = \langle x, [1,2,4]^T\rangle/\|x\|^3$, $x \notin \Omega$, as boundary conditions. Since in this case the density is unknown, we measure only the ℓ^∞-norm of the error of the solution with respect to about 8000 evaluation points x_i distributed inside the gearwheel. We emphasize that the density ρ exhibits strong singularities due to the presence of vertices and edges, even though the solution u is smooth. Hence, we cannot expect the same orders of convergence as on the sphere.

In Tables 10.1 and 10.2 we tabulated the results for the Dirichlet and the Neumann problem, respectively. The cpu-time refers to the computation of the approximate density, i.e., assembling and solving the linear system of equations by a CG or GMRES algorithm. The bracketed values correspond to the ratio of the error on the previous level to the actual error and reflect the order of convergence. The number of nonzero coefficients of the system matrix (density) is given in percent; see also Figure 10.4 for a visualization. The present example demonstrates that the wavelet Galerkin scheme remains applicable also in the case of complex geometries. It turns out that a piecewise bilinear discretization requires about twice the memory and five times the cpu-time of the corresponding piecewise constant discretization. However, the higher order yields a much higher accuracy. In Figure 10.5 we plot the absolute error versus the cpu-time. In the case of the Dirichlet problem (left-hand plot), one observes that the single layer operator discretized by piecewise bilinears yields the best performance. In the case of the Neumann problem (right-hand plot), the situation is not so clear. However, when extrapolating the graphs, the adjoint of the double layer operator discretized by piecewise bilinears seems to perform best.

TABLE 10.1
Numerical results with respect to the Dirichlet problem on the gearwheel.

Problem	J	N_J	$\|\mathbf{u}-\mathbf{u}_J\|_\infty$	Cpu-time (sec.)	Density (%)
Equation (10.1)	1	1344	6.60e-2	1	15
discretized by	2	5376	3.66e-2 (1.8)	8	5.9
piecewise constant	3	21504	4.17e-3 (8.8)	65	2.0
wavelets	4	86016	1.10e-3 (3.8)	502	0.65
	5	344064	2.68e-4 (4.1)	2693	0.18
Equation (10.2)	1	1344	3.34e-1	1	15
discretized by	2	5376	1.12e-1 (2.9)	8	3.9
piecewise constant	3	21504	2.22e-2 (5.2)	52	1.3
wavelets	4	86016	6.24e-3 (3.6)	389	0.37
	5	344064	6.34e-4 (10)	1717	0.09
Equation (10.1)	1	3024	2.84e-2	16	100
discretized by	2	8400	1.27e-2 (2.2)	32	17
piecewise bilinear	3	27216	4.91e-4 (26)	324	4.8
wavelets	4	97104	3.14e-5 (16)	2366	1.4
Equation (10.2)	1	3024	4.72e-1	10	100
discretized by	2	8400	1.02e-1 (4.6)	28	16
piecewise bilinear	3	27216	7.90e-3 (13)	309	3.4
wavelets	4	97104	1.10e-3 (7.2)	2334	0.92

TABLE 10.2
Numerical results with respect to the Neumann problem on the gearwheel.

Problem	J	N_J	$\|\mathbf{u}-\mathbf{u}_J\|_\infty$	Cpu-time (sec.)	Density (%)
Equation (10.4)	1	1344	9.51e-1	1	15
discretized by	2	5376	3.98e-1 (2.4)	8	3.9
piecewise constant	3	21504	4.31e-2 (9.2)	52	1.3
wavelets	4	86016	1.20e-2 (3.6)	389	0.37
	5	344064	2.52e-4 (4.8)	1717	0.09
Equation (10.3)	1	1344	8.59e-1	6	100
discretized by	2	5376	2.50e-1 (3.4)	88	16
piecewise bilinear	3	21504	1.44e-2 (17)	616	4.4
wavelets	4	86016	1.88e-3 (7.6)	3044	1.2
Equation (10.4)	1	3024	7.10e-1	10	100
discretized by	2	8400	2.38e-1 (3.0)	28	16
piecewise bilinear	3	27216	1.03e-2 (23)	309	3.4
wavelets	4	97104	2.34e-3 (4.4)	2343	0.92

All computations were performed by a standard workstation with 2 Gigabyte main memory. We never needed a degree of quadrature higher than 11. Moreover, we set $d' = \delta = 1.25$ if $d = 1$ and $d' = \delta = 2.25$ if $d = 2$, and $s = r = 1$. The parameter a was chosen equal to 1 for the sphere, which to our experience is a robust choice if all distances are scaled such that the largest patch has a diameter $\sqrt{2}$ that corresponds to the unit square. However, in our second example we increased this parameter to $a = 1.5$ to achieve an accuracy on the coarse grids that is comparable to that produced by the traditional Galerkin scheme. Finally, let us remark that we never subdivided elements living on the same levels, that is, $M = 0$. This choice does a good job if the constant c_Γ in the proof of Lemma 9.7 is sufficiently large, which is the case for both geometries. But we emphasize that M has to be increased if the geometry contains patches that meet in a very acute angle.

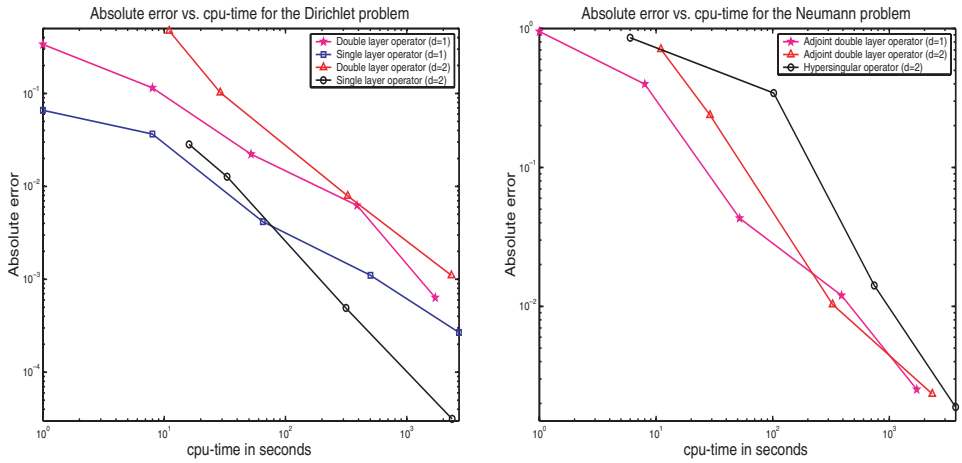

FIG. 10.5. *Performance with respect to the Dirichlet (left) and to the Neumann (right) problem.*

REFERENCES

[1] G. BEYLKIN, R. COIFMAN, AND V. ROKHLIN, *The fast wavelet transform and numerical algorithms*, Comm. Pure Appl. Math., 44 (1991), pp. 141–183.

[2] C. CANUTO, A. TABACCO, AND K. URBAN, *The wavelet element method, part* I: *Construction and analysis*, Appl. Comput. Harmon. Anal., 6 (1999), pp. 1–52.

[3] A. COHEN, I. DAUBECHIES, AND J.-C. FEAUVEAU, *Biorthogonal bases of compactly supported wavelets*, Pure Appl. Math., 45 (1992), pp. 485–560.

[4] A. COHEN AND R. MASSON, *Wavelet adaptive method for second order elliptic problems: Boundary conditions and domain decomposition*, Numer. Math., 86 (2000), pp. 193–238.

[5] W. DAHMEN, *Wavelet and multiscale methods for operator equations*, Acta Numer., 6 (1997), pp. 55–228.

[6] W. DAHMEN, H. HARBRECHT, AND R. SCHNEIDER, *Compression techniques for boundary integral equations—asymptotically optimal complexity estimates*, SIAM J. Numer. Anal., 43 (2006), pp. 2251–2271.

[7] W. DAHMEN AND A. KUNOTH, *Multilevel preconditioning*, Numer. Math., 63 (1992), pp. 315–344.

[8] W. DAHMEN, B. KLEEMANN, S. PRÖSSDORF, AND R. SCHNEIDER, *A multiscale method for the double layer potential equation on a polyhedron*, in Advances in Computational Mathematics, Ser. Approx. Decompos. 4, H. P. Dikshit and C. A. Micchelli, eds., World Scientific, River Edge, NJ, 1994, pp. 15–57.

[9] W. DAHMEN, A. KUNOTH, AND K. URBAN, *Biorthogonal spline-wavelets on the interval—stability and moment conditions*, Appl. Comput. Harmon. Anal., 6 (1999), pp. 132–196.

[10] W. DAHMEN, S. PRÖSSDORF, AND R. SCHNEIDER, *Wavelet approximation methods for pseudodifferential equations*. II. *Matrix compression and fast solution*, Adv. Comput. Math., 1 (1993), pp. 259–335.

[11] W. DAHMEN, S. PRÖSSDORF, AND R. SCHNEIDER, *Multiscale methods for pseudodifferential equations on smooth closed manifolds*, in Wavelets: Theory, Algorithms, and Applications (Taormina, 1993), Wavelet Anal. Appl. 5, Academic Press, San Diego, 1994, pp. 385–424.

[12] W. DAHMEN AND R. SCHNEIDER, *Composite wavelet bases for operator equations*, Math. Comp., 68 (1999), pp. 1533–1567.

[13] W. DAHMEN AND R. SCHNEIDER, *Wavelets on manifolds* I: *Construction and domain decomposition*, SIAM J. Math. Anal., 31 (1999), pp. 184–230.

[14] W. DAHMEN AND R. STEVENSON, *Element-by-element construction of wavelets satisfying stability and moment conditions*, SIAM J. Numer. Anal., 37 (1999), pp. 319–352.

[15] M. G. DUFFY, *Quadrature over a pyramid or cube of integrands with a singularity at a vertex*. SIAM J. Numer. Anal., 19 (1982), pp. 1260–1262.

[16] L. GREENGARD AND V. ROKHLIN, *A fast algorithm for particle simulation*, J. Comput. Phys., 73 (1987), pp. 325–348.

[17] W. HACKBUSCH AND B. N. KHOROMSKIJ, *A sparse $\mathcal{H}$-matrix arithmetic*. II. *Application to*

multi-dimensional problems, Computing, 64 (2000), pp. 21–47.
[18] W. Hackbusch and Z. P. Nowak, *On the fast matrix multiplication in the boundary element method by panel clustering*, Numer. Math., 54 (1989), pp. 463–491.
[19] G. Hämmerlin and K.-H. Hoffmann, *Numerical Mathematics*, Springer-Verlag, New York, 1991.
[20] H. Harbrecht, *Wavelet Galerkin Schemes for the Boundary Element Method in Three Dimensions*, Ph.D. Thesis, Technische Universität Chemnitz, Chemnitz, Germany, 2001.
[21] H. Harbrecht, F. Paiva, C. Pérez, and R. Schneider, *Biorthogonal wavelet approximation for the coupling of FEM-BEM*, Numer. Math., 92 (2002), pp. 325–356.
[22] H. Harbrecht, F. Paiva, C. Pérez, and R. Schneider, *Wavelet preconditioning for the coupling of FEM-BEM*, Numer. Linear Algebra Appl., 10 (2003), pp. 197–222.
[23] H. Harbrecht and R. Schneider, *Biorthogonal wavelet bases for the boundary element method*, Math. Nachr., 269/270 (2004), pp. 167–188.
[24] K. Höllig and H. Mögerle, *G-splines*, Comput. Aided Geom. Design, 7 (1990), pp. 197–207.
[25] J.-C. Nedelec, *Integral equations with nonintegrable kernels*, Integral Equations Operator Theory, 5 (1982), pp. 562–572.
[26] T. von Petersdorff, C. Schwab, and R. Schneider, *Multiwavelets for second-kind integral equations*, SIAM J. Numer. Anal., 34 (1997), pp. 2212–2227.
[27] T. von Petersdorff and C. Schwab, *Fully discrete multiscale Galerkin BEM*, in Multiscale Wavelet Methods for PDEs, W. Dahmen, A. Kurdila, and P. Oswald, eds., Academic Press, San Diego, 1997, pp. 287–346.
[28] U. Reif, *Biquadratic G-spline surfaces*, Comput. Aided Geom. Design, 12 (1995), pp. 193–205.
[29] S. Sauter, *Über die effiziente Verwendung des Galerkin Verfahrens zur Lösung Fredholmscher Integralgleichungen*, Ph.D. Thesis, Christian–Albrecht–Universität Kiel, Kiel, Germany, 1992.
[30] S. Sauter and C. Schwab, *Quadrature for the hp-Galerkin BEM in* $\mathbb{R}^3$, Numer. Math., 78 (1997), pp. 211–258.
[31] R. Schneider, *Multiskalen- und Wavelet-Matrixkompression: Analysisbasierte Methoden zur Lösung großer vollbesetzter Gleichungssysteme*, B.G. Teubner, Stuttgart, Germany, 1998.
[32] C. Schwab, *Variable order composite quadrature of singular and nearly singular integrals*, Computing, 53 (1994), pp. 173–194.
[33] L. F. Villemoes, *Wavelet analysis of refinement equations*, SIAM J. Math. Anal., 25 (1994), pp. 1433–1460.
[34] W. L. Wendland, *On asymptotic error analysis and underlying mathematical principles for boundary element methods*, in Boundary Element Techniques in Computer Aided Engineering, NATO ASI Series E - 84, C. A. Brebbia, ed., Martinus Nijhoff, Dordrecht, Boston, Lancaster, 1984, pp. 417–436.