

**A two-dimensional
wavelet-packettransform for matrix
compression of integral equations with
highly oscillatory kernel**

Daan Huybrechs and Stefan Vandewalle

Report TW 417, January 2005 (revised August 2005)



Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

A two-dimensional wavelet-packettransform for matrix compression of integral equations with highly oscillatory kernel

Daan Huybrechs and Stefan Vandewalle

Report TW417, January 2005 (revised August 2005)

Department of Computer Science, K.U.Leuven

Abstract

We examine the use of wavelet packets for the fast solution of integral equations with a highly oscillatory kernel. The redundancy of the wavelet packet transform allows the selection of a basis tailored to the problem at hand. It is shown that a well chosen wavelet packet basis is better suited to compress the discretized system than wavelets. The complexity of the matrix-vector product in an iterative solution method is then substantially reduced. A two-dimensional wavelet packet transform is derived and compared to a number of one-dimensional transforms that were presented earlier in literature. By means of some numerical experiments we illustrate the improved efficiency of the two-dimensional approach.

Keywords : integral equation, wavelet packets, high frequency

AMS(MOS) Classification : Primary : 35J05, Secondary : 42C40, 65N38.

A two-dimensional wavelet packet transform for matrix compression of integral equations with highly oscillatory kernel*

Daan Huybrechs and Stefan Vandewalle

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A, B-3001 Leuven, Belgium.
{Daan.Huybrechs,Stefan.Vandewalle}@cs.kuleuven.be

Abstract

We examine the use of wavelet packets for the fast solution of integral equations with a highly oscillatory kernel. The redundancy of the wavelet packet transform allows the selection of a basis tailored to the problem at hand. It is shown that a well chosen wavelet packet basis is better suited to compress the discretized system than wavelets. The complexity of the matrix-vector product in an iterative solution method is then substantially reduced. A two-dimensional wavelet packet transform is derived and compared with a number of one-dimensional transforms that were presented earlier in literature. By means of some numerical experiments we illustrate the improved efficiency of the two-dimensional approach.

1 Introduction

An integral equation formulation is frequently used for the solution of exterior boundary value problems. The domain of the unknown is then restricted to a finite boundary. This reduces the problem dimension by one in comparison to formulations based on partial differential equations. Various fast methods have been proposed for the numerical solution of such integral equation problems with $O(N \log N)$ or even $O(N)$ complexity, where N is the number of unknowns in a boundary element discretization. Among these are the Fast Multipole Method [18], methods based on Panel Clustering or Hierarchical Matrices [19], and wavelet based methods [4, 11]. In this article, we will consider the wavelet approach and apply that method for solving oscillatory problems.

It is well-known that matrix compression methods based on wavelets deteriorate for high frequency problems. Indeed, the compressed wavelet transformed discretization matrix loses its sparsity with increasing frequency. The proportionality constant in the asymptotic complexity estimate $O(N)$ is therefore a function of the frequency. For the two-dimensional Helmholtz problem, this constant grows almost linearly with the wavenumber k [27, 22]. It is common to choose N proportional to k , yielding an $O(N^2)$ complexity for high frequency problems.

*This technical report was accepted for publication in J. Comput. Appl. Math., 2005.

The goal of this paper is to describe a fast solution method for the two-dimensional Helmholtz equation with a wavenumber that is large with respect to the size of the boundary. We use an integral equation formulation of the problem and discretize that equation by using boundary elements. Our aim is to solve the corresponding discretized system of equations with an iterative Krylov-space solver. The matrix-vector product will be accelerated by a sparsifying transformation of the discretization matrix based on using wavelet packets. Wavelet packets exhibit a finer frequency resolution than wavelets, and it is shown that they are better suited to compress problems with a certain inherent frequency. This was reported earlier by Deng and Ling [15, 16] and by Golik [17]. Here, we propose a different wavelet packet transform, and show that the results of the above references can be further improved, quite considerably. We present and motivate the model formulation in Section 2. Some essential theory of wavelets and wavelet packets is recalled in Section 3. The application of wavelet packets to integral equations is discussed in Section 4. Numerical results are presented in Section 5, that demonstrates the matrix compression performance of the wavelet packet basis. We analyze the improved computational complexity in Section 6, and we compare with competitive methods. Finally, we end in Section 7 with some concluding remarks.

2 Model formulation

2.1 The combined field integral equation for the Helmholtz problem

We consider the two-dimensional Helmholtz equation

$$\Delta u + k^2 u = 0, \quad (1)$$

defined on the complement in \mathbb{R}^2 of a compact and connected region Ω , with Dirichlet or Neumann boundary conditions imposed on the boundary of Ω , i.e., the closed curve Γ . There exist several equivalent formulations of this problem as an integral equation. In this section, we point out the formulation that will be used throughout the article.

Four basic integral operators are widely used in the solution of elliptic boundary value problems: the single layer potential S , the double layer potential D and its adjoint D^* , and the hypersingular operator N . These operators are described for the Laplace equation and the Helmholtz equation in, e.g., [25], both for the two-dimensional and the three-dimensional case. A review of the relevant theory can be found in [24]. The kernel function for the two-dimensional Helmholtz problem is given by

$$K(x, y) := \frac{i}{4} H_0^{(1)}(k|x - y|),$$

with $H_0^{(1)}(z)$ the Hankel function of the first kind of order zero and i the imaginary unit. The above mentioned integral operators are given by

$$\begin{aligned} (Su)(x) &:= \int_{\Gamma} K(x, y) u(y) ds_y & (D^*u)(x) &:= \int_{\Gamma} \frac{\partial K}{\partial \nu(x)}(x, y) u(y) ds_y \\ (Du)(x) &:= \int_{\Gamma} \frac{\partial K}{\partial \nu(y)}(x, y) u(y) ds_y & (Nu)(x) &:= \frac{\partial}{\partial \nu(x)} \int_{\Gamma} \frac{\partial K}{\partial \nu(y)}(x, y) u(y) ds_y \end{aligned}$$

The notation $\frac{\partial}{\partial\nu(x)}$ and $\frac{\partial}{\partial\nu(y)}$ is used to denote the outward normal derivative with respect to the x -variable and the y -variable respectively. The use of these operators leads to the following four integral equation formulations of the exterior boundary value problem,

$$Sq = f \quad \left(\frac{I}{2} + D^*\right)q = g \quad (2)$$

$$\left(-\frac{I}{2} + D\right)r = f \quad -Nr = g \quad (3)$$

Here, f represents the given Dirichlet boundary condition's right hand side function, and g the Neumann boundary condition's right hand side function. The functions q and r are the densities of the single layer potential and the double layer potential, respectively.

The solution of the exterior Dirichlet and Neumann boundary value problems is unique. However, the integral formulations (2) and (3) become singular for the countably infinite set of values of the wavenumber k that correspond to the eigenvalues of the Laplacian in the interior domain. This singularity can be overcome by combining the formulations. A common formulation in electromagnetics is the so-called *Combined Field Integral Equation*. This equation is a linear combination of the equations in (2), and is given by

$$i\omega\mu_0\left(S + i\alpha\left(\frac{I}{2} + D^*\right)\right)q = i\omega\mu_0(f + i\alpha g). \quad (4)$$

The scaling factor $i\omega\mu_0$, or, equivalently, $ick\mu_0$, with c the speed of light and μ_0 the permeability of free space, is added so that the solution q has a physical meaning. It represents the current on Γ induced by an incoming electromagnetic wave. Equation (4) is valid for all values of the wavenumber whenever $\alpha \neq 0$. The parameter α in (4) can be chosen freely. For the purpose of this article we will choose it such that the condition number of the corresponding discretized system of equations is minimized. This was investigated by Amini [2], and leads to the choice

$$\alpha = \frac{1}{k}. \quad (5)$$

The value of α has a large effect on the condition number of the discrete problem, and thus also on the number of matrix-vector products that are required in an iterative solution method. We will quantify the difference with the common choice $\alpha = 1$ by means of some numerical results in Section 5.

2.2 Collocation and Galerkin approaches for discretization

We will consider two different discretization schemes for equation (4): the collocation and Galerkin methods. For the collocation approach we partition Γ into N segments or elements. A set of pulse basis functions is used, each with height 1 on one element and zero elsewhere. We apply a one point integration formula for the ds_y -integrals, as given in [20]. Define Δ_i , r_i and n_i as the width, center position, and outgoing normal of the i th pulse. The discretization matrix M is then given by

$$M_{i,j} = \begin{cases} i\omega\mu_0\Delta_j\frac{i}{4}\left(1 + \frac{2i}{\pi}\log\left(\frac{e^{\gamma}k\Delta_j}{4e}\right)\right) - \omega\mu_0\frac{\alpha}{2} & \text{if } i = j \\ i\omega\mu_0\Delta_j\left(\frac{i}{4}H_0^{(1)}(k|r_i - r_j|) - i\alpha\left(n_i \cdot \frac{r_i - r_j}{|r_i - r_j|}\right)k\frac{i}{4}H_1^{(1)}(k|r_i - r_j|)\right) & \text{otherwise.} \end{cases} \quad (6)$$

We also implemented a full Galerkin method, where both the trial and test functions are the scaling functions ϕ_i of Daubechies wavelets [13]. The elements are given by

$$M_{i,j} = \langle i\omega\mu_0(S + i\alpha(\frac{I}{2} + D^*))\phi_j, \phi_i \rangle. \quad (7)$$

They can efficiently be computed by using the integration routines described in [23].

The approximation to the solution of (4) is found by solving

$$Mx = b, \quad (8)$$

with the elements of M given by (6) or (7) and with b the corresponding discretization of the right hand side of (4) on Γ . Typically, the size N of matrix M is chosen proportional to k . In our numerical experiments, for example, we take 10 elements per wavelength. For solving (8) iterative methods of Krylov subspace type can be very effective. Such methods are based on matrix-vector products. The computational complexity of one matrix-vector product is $O(N^2)$. Here, the matrix M will be transformed and compressed in order to obtain a faster matrix-vector product. Several matrix compression techniques will be compared in this paper. The compression error can be measured in different ways. Assume x is the exact solution of (8), and y is the solution of the compressed problem. Then the relative error of the solution and the relative residual error are given by

$$e_S = \frac{\|x - y\|}{\|x\|} \quad \text{and} \quad e_R = \frac{\|b - My\|}{\|b\|}, \quad (9)$$

respectively. The residual error is a weaker error measure, but it is very useful in practice. E.g., in computational electromagnetics applications an approximate solution y with residual error e_R represents a current that induces the same electromagnetic field as the exact solution x to a certain precision e_R . Both error measures will be considered in the numerical examples.

3 A review of wavelets and wavelet packets

3.1 Wavelets

The theory of wavelets is described, e.g., in [14]. We will repeat some basic facts here mainly to establish notation. First, we recall the two scale relation satisfied by the scaling function ϕ , and we define the corresponding wavelet function ψ as follows,

$$\phi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_k \phi(2x - k) \quad \text{and} \quad \psi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} g_k \phi(2x - k)$$

with suitable choices of coefficients g_k and h_k . The wavelets have a certain number d of vanishing moments, i.e., $\int \psi(x)x^l dx = 0$, for $l = 0, \dots, d - 1$.

Wavelets enable a multiresolution analysis. Define the shifted and dilated wavelet and scaling functions as $\phi_{jk}(t) := 2^{j/2}\phi(2^j t - k)$ and $\psi_{jk}(t) := 2^{j/2}\psi(2^j t - k)$. Then the corresponding function spaces $V_j := \text{span}\{\phi_{jk}(t)\}$ and $W_j := \text{span}\{\psi_{jk}(t)\}$ on scale j satisfy

$$V_0 \subset V_1 \subset V_2 \subset \dots \subset V_j \subset \dots \quad \text{and} \quad V_j = V_{j-1} \oplus W_{j-1}.$$

This leads to the decomposition

$$V_J = V_0 \oplus_{j=0}^{J-1} W_j. \quad (10)$$

Any function $f \in V_J$ can be expanded in the basis of scaling functions $\{\phi_{Jk}(t)\}$ or in the basis suggested by (10), i.e.,

$$f = \sum_k v_{Jk} \phi_{Jk} \quad \text{and} \quad f = \sum_k v_{0k} \phi_{0k} + \sum_{j=0}^{J-1} \sum_k w_{jk} \psi_{jk}. \quad (11)$$

One can go from one representation to the other by using the Fast Wavelet Transform, that is based on a repeated application of the following relations,

$$v_{jk} = \sum_l h_{l-2k} v_{j+1,l} \quad \text{and} \quad w_{jk} = \sum_l g_{l-2k} v_{j+1,l}. \quad (12)$$

Starting at a scale $j+1$ with scaling function coefficients $v_{j+1,k}$, application of (12) gives the values v_{jk} and w_{jk} such that

$$\sum_k v_{j+1,k} \phi_{j+1,k}(t) = \sum_k v_{jk} \phi_{jk}(t) + \sum_k w_{jk} \psi_{jk}(t).$$

The inverse transform is performed by using the formula

$$v_{j+1,k} = \sum_l h_{k-2l} v_{jl} + \sum_l g_{k-2l} w_{jl}. \quad (13)$$

The computational complexity of the Fast Wavelet Transform and its inverse for a function in V_J is $O(N)$, with N the dimension of V_J . Both transforms can conceptually be written as a matrix vector product, with a transformation matrix $T \in \mathbb{R}^{N \times N}$ and its inverse T^{-1} .

Decomposition (10) divides the frequency spectrum of the original function space. The scaling functions on the coarse scale capture the lowest frequency content of a function, while the wavelets progressively add more detail and higher frequency content. The basis functions on coarser scales are more localized in frequency, at a cost of spatial resolution, while the basis functions on finer scales are more localized in space, at a cost of frequency resolution.

3.2 Wavelet packets

Wavelet packets can be seen as a generalization of wavelets. They were introduced and developed by Coifman, Meyers, Quake and Wickerhauser [8, 10]. The idea is based on additional applications of the so-called splitting trick. In the wavelet transform, function space V_{j+1} is split into V_j and W_j , using the filters g and h and the relations (12). The resulting space V_j is split again, a process that is continued until decomposition (10) is obtained.

One can also split function space W_j further by using the same filters. Two function spaces are then obtained with direct sum equal to W_j . Continued recursive application leads to a binary tree of function spaces with root V_J . The basis functions of these spaces are called wavelet packets. The frequency resolution increases downwards in the tree, now also for the higher frequencies, again at the cost of spatial resolution. What part of the frequency spectrum lies in each function space is the subject of [21]. The trees for the wavelet and wavelet packet function spaces are shown in Figure 1.

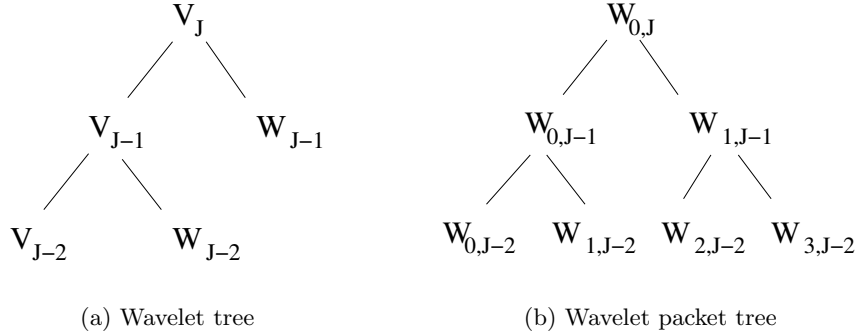


Figure 1: Tree representation of wavelet and wavelet packet function spaces.

To describe this more formally, we introduce a parameter n to denote the frequency. Set $w_{0,0,0} = \phi$ and define recursively

$$w_{2n,0,0}(t) = \sqrt{2} \sum_k h_k w_{n,0,0}(2t - k) \quad \text{and} \quad w_{2n+1,0,0}(t) = \sqrt{2} \sum_k g_k w_{n,0,0}(2t - k). \quad (14)$$

The wavelet packets are now given by $w_{njk}(t) = 2^{j/2} w_{n00}(2^j t - k)$. We represent the function spaces involved by $W_{nj} = \text{span}\{w_{njk}\}$. A basis of $V_J = W_{0J}$ can be identified by a subset Λ of the set of indices $\Xi := \{(n, j) \in \mathbb{Z}^2\}$, such that the corresponding wavelet packets w_{njk} form a basis of V_J . It is convenient here to use the multi-index notation $\lambda = (n, j)$. We can expand any function $f \in V_J$ in the basis denoted by Λ as

$$f = \sum_{\lambda \in \Lambda} \sum_k v_{\lambda,k} w_{\lambda,k}. \quad (15)$$

Both decompositions of f in (11) are special cases of (15). A fast wavelet packet transformation can be devised, by using relations similar to (12). The full wavelet packet decomposition at level J is the transform corresponding to the basis functions w_{njk} . The computational complexity of this transformation is $O(J2^J) = O(N \log N)$.

A two-dimensional wavelet packet transform of a matrix A can be defined by applying a one-dimensional transform to all rows and columns of the matrix successively. The resulting matrix is called the rectangular transform of A . Alternatively, a two-dimensional wavelet packet transform can be obtained by considering a quadtree of function spaces of the form $W_\lambda \times W_\mu$, for $\lambda, \mu \in \Xi \times \Xi$. A subtree with any selection of function spaces that covers $V_J \times V_J$ leads to a two-dimensional basis. The tensor product basis functions are given by

$$w_{\mu,l,\lambda,k}(s, t) = w_{\mu,l}(s) w_{\lambda,k}(t). \quad (16)$$

This approach is called the square transform. The structure of the square transform B of a matrix is shown in the left hand side of Figure 2(a). We denote each square subblock by $B_{\mu,\lambda}$.

3.3 Best basis algorithm of Coifman and Wickerhauser

It can be proven that, for a vector x of N elements, there exist more than 2^N possible wavelet packet bases. Coifman and Wickerhauser presented a method to find a best basis for a given

criterion [10]. The algorithm finds a global minimum for a cost function $P(\{x_\lambda\})$, among all possible wavelet packet representations $\{x_\lambda\}$ of x . It is applicable for cost functions that satisfy $P(\emptyset) = 0$, and $P(\{t_i\}) = \sum_i p(|t_i|)$ for some function p , i.e., it is additive. The choice

$$p(t) = \begin{cases} 1 & \text{if } |t| > \tau \\ 0 & \text{otherwise.} \end{cases}$$

leads to the wavelet packet transform with the smallest number of elements larger than τ .

The algorithm uses a bottom-up approach. First, a full wavelet-packet decomposition of x is computed up to scale 0. The lowest cost representation of the part of x that lies in $W_{0,1} = W_{0,0} \oplus W_{1,0}$ has an associated cost

$$Q(0, 1) := \min\{P(\{x_{0,1,k}\}), P(\{x_{0,0,k}\}) + P(\{x_{1,0,k}\})\}. \quad (17)$$

A similar expression can be given for $Q(n, 1)$, $n = 1 \dots 2^{J-1}$. The lowest cost representation of the part of x that lies in $W_{n,j}$, $j > 1$, is given by

$$Q(n, j) := \min\{P(\{x_{n,j,k}\}), Q(2n, j-1) + Q(2n+1, j-1)\}. \quad (18)$$

By induction, the global minimum of P is given by $Q(0, J)$. The corresponding best basis is found by remembering the arguments that minimize the expressions (17) and (18).

An extension of this algorithm to the two-dimensional case is possible only for the square transform discussed previously. The cost of a subblock $B_{\lambda,\mu}$ is evaluated by $P(B_{\lambda,\mu})$. An additive cost function cannot be found for the rectangular transform, since in that case the subblocks corresponding to λ and μ , for $\lambda, \mu \in \Lambda$, overlap for each combination of λ and μ . Hence, in that case the efficient best basis algorithm can only be approximated.

4 Wavelet packets for the solution of integral equations

4.1 Earlier work

The use of wavelet packets for the fast solution of integral equations has been considered previously in [17, 15, 16]. Deng and Ling, and Golik independently studied wavelet packet based matrix compression. Both reported a number of significant elements after compression that scales as $O(N^{4/3})$ for the Combined Field Integral Equation (4) with collocation. Both approaches were based on a one-dimensional transform and, hence, only an approximation to the best basis algorithm was applied. In [17] a top-down approximation to the best basis algorithm is performed on the right hand side of the linear system (8). The resulting basis is used for compressing the matrix using the rectangular transform. In [15] a top-down approximation to the best basis for the rectangular transform is performed on the matrix in (8) itself. In [16] a one-dimensional wavelet packet basis is constructed that zooms in on the frequency given by k [16].

Here, we will consider the use of a two-dimensional wavelet packet basis using the square transform. This will increase the freedom in the choice of basis greatly. Moreover, the best basis algorithm can be applied exactly and the sparsity results can be much improved.

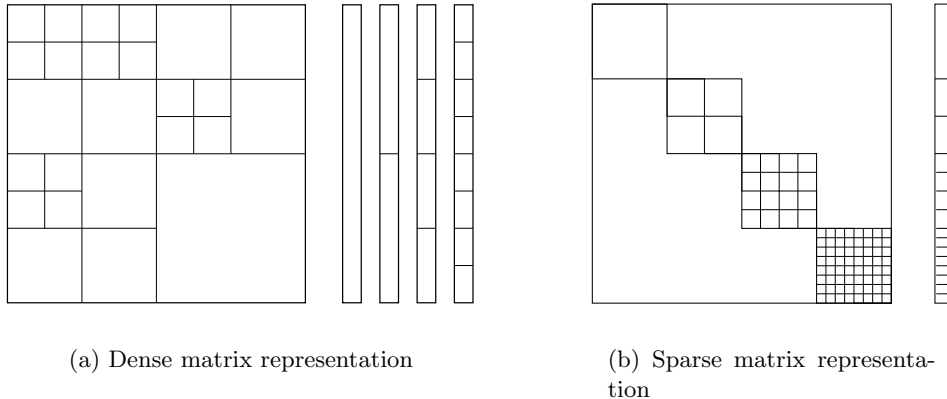


Figure 2: Two different implementations of a matrix-vector product Mx with a wavelet-packet transformed matrix. The full wavelet-packet transformation on each scale of vector x is shown side by side in (a), and as one large vector in (b).

4.2 The nonstandard matrix-vector product

A one-dimensional wavelet packet transform U of a regular matrix M can be written as $U = TMT'$. The matrix-vector product $y = Mx$ can then be computed easily as $y = T^{-1}U(T')^{-1}x$. A matrix-vector product with a matrix obtained after a two-dimensional wavelet packet transformation is more involved, but can still be defined [28]. The algorithm for the matrix-vector product is a three-step procedure. Assume that W is a two-dimensional wavelet packet transform of M . First, the representation coefficients x_λ , for $\lambda \in \Xi$ are computed for all scales $j = 0, \dots, J$. Next, for each block $W_{\mu,\lambda}$ in W that corresponds to a function space $W_\mu \times W_\lambda$, compute

$$y_\mu := W_{\mu,\lambda}x_\lambda.$$

Finally, the result y is given by

$$y = \sum_{\mu} \sum_k y_{\mu,k} \psi_{\mu,k}.$$

The latter is easily computed by adding the inverse transformation of each y_μ . The algorithm is clarified in Figure 2(a). The right hand side of the figure depicts the representations of x on each scale j . The left hand side of the figure shows the structure of the transformed matrix with subblocks $W_{\mu,\lambda}$. The matrix-vector product requires one to multiply each subblock on scale j of the dense matrix with the matching block on scale j of x . Note that the structure of the matrix strongly resembles that of the so-called H^2 -matrices used in the fast solution of integral equations [5]. The matrix-vector product is also very analogous. The three steps described above relate directly to the three steps for a matrix-vector product with H^2 -matrices: Forward Transformation, Multiplication, and Backwards Transformation (see [5]).

The algorithm as described above requires many small matrix-vector products and the construction of special purpose software. It is possible however to simplify the implementation considerably, by noting that the computations can be rewritten as one matrix-vector product with a large very sparse matrix. Consider a block-diagonal matrix S of total dimension

$N \log_2 N \times N \log_2 N$, where each of the $\log_2 N$ subblocks of size $N \times N$ on the diagonal corresponds to the full wavelet-packetdecomposition of M at a scale j , for $j = 0 \dots \log_2 N - 1$. (Note that there will be no need to construct S entirely.) Define the extended vector e as the $N \log_2 N \times 1$ vector $(x_1, \dots, x_j, \dots, x_{\log_2 N})^T$, where each subvector x_j of N elements corresponds to the full wavelet packet decomposition of x at a scale j . Matrix S and vector e are depicted in Figure 2(b). The matrix S and the vector e contain all the information that is needed to compute the matrix-vector product for any possible two-dimensional wavelet packet basis by the following algorithm.

1. Compute the extended vector e .
2. Construct the sparse matrix S' by copying from S only those subblocks that correspond to a function space $W_\mu \times W_\lambda$ that is in the wavelet packet quadtree.
3. Compute the regular matrix-vector product $u = S'e$.
4. Sum the inverse wavelet-packet transformation of each component of u .

The result is exactly the same as in the previous algorithm. However, the new algorithm is easier to implement since no special data structures are required. Many existing codes already implement optimized matrix-vector products for large, sparse matrices, sequentially or in parallel.

4.3 Matrix compression: scaling of the threshold

A compression of the discretization matrix is obtained by discarding small elements in the transformed matrix. The most straightforward implementation uses a fixed threshold value τ , i.e., the compressed matrix W^ϵ is given by

$$W_{i,j}^\epsilon = \begin{cases} W_{i,j} & \text{if } |W_{i,j}| > \tau \\ 0 & \text{otherwise.} \end{cases} \quad (19)$$

A more advanced compression strategy utilizes a scale dependent threshold [11]. Such a strategy can improve sparsity, but for the arguments in this paper the simpler strategy is already sufficient. A natural question that arises is how to choose the parameter τ as a function of N or k , in order to guarantee a fixed error. To answer this question, we will construct an estimate for the matrix compression error as a function of τ and N (or k). We will consider the matrix obtained with the collocation scheme, and we measure the matrix compression error as the relative error

$$e = \frac{\|W - W^\epsilon\|_2}{\|W\|_2}. \quad (20)$$

First we determine the asymptotic behaviour of $\|M\|_2$, with M the collocation matrix whose elements are given in (6). The Hankel functions for large arguments behave as

$$|H_i^{(1)}(z)| \sim \frac{1}{\sqrt{z}},$$

independent of the order i [1]. The pulse width Δ_j is obviously $O(1/N) = O(h)$, and $\omega = ck = O(k)$. The contribution of the single layer potential to the off-diagonal elements in (6) is therefore $O(hk/\sqrt{k})$. The contribution of D^* is $O(hk^2/\sqrt{k})$. With the optimal choice of $\alpha = 1/k$, the linear combination is also of order $O(hk/\sqrt{k})$. The diagonal element contributions from S and D^* are respectively $O(h \log k)$ and $O(1)$, so the linear combination is $O(1)$.

The maximum absolute column sum norm $\|M\|_1$ and the maximum absolute row sum norm $\|M\|_\infty$ are of order $O(1 + (N - 1)(hk/\sqrt{k})) = O(Nhk/\sqrt{k}) = O(\sqrt{k})$. From this and from the inequality $\|M\|_2^2 \leq \|M\|_1 \|M\|_\infty$ we can deduce that

$$\|M\|_2 = O(\sqrt{k}). \quad (21)$$

The orthogonal wavelet transform and orthogonal wavelet packet transform can be represented by a transformation matrix T with $\|T\|_2 = 1$. Therefore, $\|W\|_2 = \|M\|_2$.

The error $\|W - W^\epsilon\|_1$ can be bounded by the worst case value $N\tau$. In fact, noting that the number of nonzero elements in W^ϵ grows approximately linearly in k [27, 22], the number of discarded elements is also proportional to k and $\|W - W^\epsilon\|_1$ could be bounded by $ak\tau$ for some real constant $a > 0$. This will not make a difference compared to the worst case value $N\tau$, because for our purposes $k \sim N$. The infinity norm is similar, leading to $\|W - W^\epsilon\|_2 = O(N\tau)$. Combined with (21), this yields

$$e = \frac{\|W - W^\epsilon\|_2}{\|W\|_2} = O(N\tau/\sqrt{k}).$$

The error (20) will be bounded only if

$$\tau = O(N^{-1/2}). \quad (22)$$

Remark 1. A threshold that is often proposed in the literature is $\tau = \frac{\|M\|_1}{N}$ [20]. This threshold has indeed the appropriate asymptotic behaviour.

Remark 2. The analysis for the Galerkin matrix with elements given by (7) is largely the same. The integration with the test function ϕ_i would lead to an additional factor $\Delta_i = O(1/N)$ in a one-point quadrature rule, when compared to (7). This is compensated by the product of the two scaling factors $2^{J/2}$ of the trial and test functions ϕ_j and ϕ_i , since $2^J = N$. The scaling factors arise from the expression $\phi_{Jk}(x) = 2^{J/2}\phi(2^J x - k)$. Hence, the size of the elements of (7) and (6) is of the same order, and the analysis above leads to the same asymptotic behaviour (22) for the threshold in the Galerkin method.

4.4 Choice of wavelet family

The domain in the setting of this article is a closed curve Γ . We can therefore apply periodic wavelets and, by doing so, avoid any edge effects at domain boundaries. This allows us to concentrate solely on the compression effect.

It can be shown that an optimal implementation of the wavelet method, i.e., one that has a computational complexity of $O(N)$, should employ biorthogonal, rather than orthogonal wavelets [11]. Wavelet packets based on biorthogonal wavelet filters are not guaranteed to be stable however [7], so we are restricted to orthogonal wavelet packets. We employ the popular Daubechies wavelets since they are compactly supported. Other orthogonal wavelets with finite filters would lead to similar results.

The compression is influenced by the number of vanishing moments d . A larger value of d leads initially to better compression, but also requires a larger number of filter coefficients h_k and g_k . This increases the computation time of the wavelet and wavelet packet transformations. A suitable tradeoff for the purposes of this article proved to be the choice $d = 7$.

4.5 Total computational complexity

Three phases of the solution method as described in this article contribute to the computational complexity: the setup of the discretization matrix, the wavelet packet transform and thresholding, and the iterative solution of the resulting system. The construction of the full discretization matrix requires $O(N^2)$ operations, both for the collocation approach and the galerkin approach. The computational complexity of the best basis algorithm for a matrix is $O(N^2 \log N)$, slightly larger than the setup cost. The complexity of the solution phase with an iterative method depends on the condition number of the matrix, and on the complexity of the matrix-vector product. The condition number is not significantly influenced by the compression, so the gain of our method lies solely in a faster matrix-vector product. The higher cost for the transformation will be compensated if the same system is solved for several different boundary conditions, since the setup has to happen only once. This is a common case e.g. in electromagnetical engineering, where incoming waves from different angles lead to different boundary conditions.

The high transformation cost can be avoided by approximating the best-basis algorithm with lower complexity methods. To this end, we considered a two-dimensional top-down approach: in each step of the wavelet packet transformation, the sparsity of a subblock in the matrix corresponding to a function space is compared to the sparsity in the representation of its four children spaces. If the sparsity is not improved, the subblock is not further transformed. The sparsity of the resulting basis is only a local minimum in the space of all possible bases, compared to the global minimum obtained by the best basis algorithm. However, the costly full wavelet packet decomposition is not required with this approach.

The high setup cost can also be avoided if the size of the elements in a wavelet packet basis can be estimated a priori. A suitable basis can then be selected that leads to a large number of small elements that do not need to be computed. Such estimates are available in the regular wavelet approach, and are based on the vanishing moments of the wavelets and the smoothness of the kernel function. Similar estimates for wavelet packets are not yet available. They should also incorporate the frequency resolution of the wavelet packets and the inherent frequency of the kernel function.

5 Numerical results

In this section, we will discuss the results of four numerical experiments. First, we evaluate the performance of the discussed matrix compression methods for a fixed threshold that scales with N as determined in Section 4.3. Next, we evaluate the matrix compression for a threshold that is adaptively chosen in order to obtain a fixed error. Third, we investigate the complexity of the number of nonzero entries in the compressed matrices, and finally we discuss the condition number of the model formulation.

We consider two different shapes for the boundary Γ , a circle and a duct, inspired by the

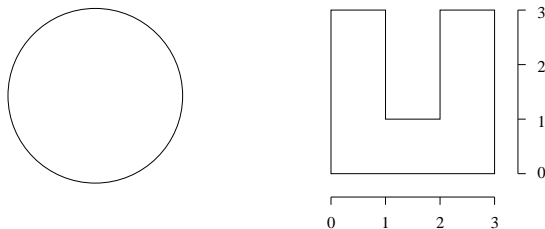


Figure 3: The obstacles: a circle with diameter $D = 1$, and a duct with circumference 16.

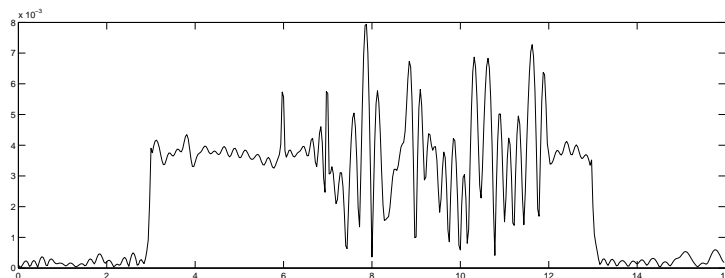
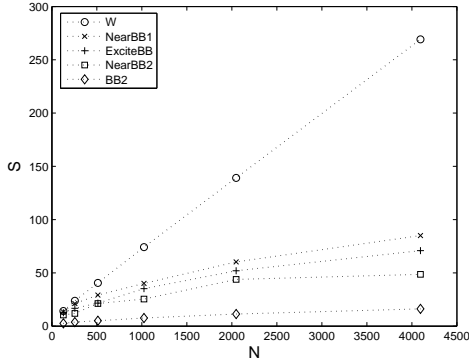


Figure 4: Absolute value of the solution along the boundary of the duct, counter-clockwise from the bottom-left point, with $k = 20$ and $N = 512$ elements.

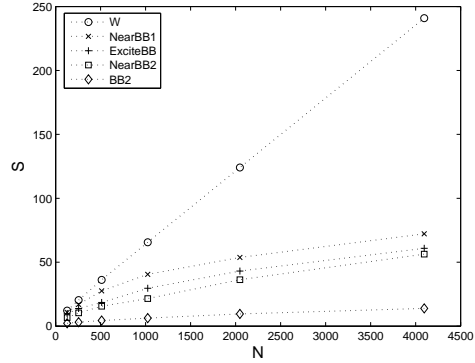
shapes used in [15, 27], see Figure 3. The circle has diameter $D = 1$, and the duct has a total circumference $D = 16$. We have obtained qualitatively similar results with other shapes. The examples are only model problems; they are used mainly to illustrate the method and its performance. Although a simple geometry, the duct has sharp corners, and a non-convex shape with possible resonances - these are two important complications in two-dimensional wave scattering. The extension of the method to more complicated geometries, including multiple scattering configurations, is straightforward. Applications include the scattering of an incoming electromagnetic wave on a wire grid (a number of cylinders with circular cross section).

The boundary condition in all examples is a plane wave e^{ikx} , incoming at an angle of 45 degrees with respect to the horizontal cross-section of the boundary. The value of k is chosen proportional to N , such that there are 10 degrees of freedom for each wavelength. For completeness, we also depict a solution for one particular value of k in Figure 4. We compare five compression methods for each example: the regular wavelet transformation itself, the approach of [15] and [17], discussed in Section 4.1, the two-dimensional best basis algorithm and its top-down approximation. These methods will be referenced respectively by (W), (NearBB1), (RhsBB), (BB2) and (NearBB2). All results were similar for the collocation and the Galerkin approach (7), so we proceed here with only the simpler discretization by collocation (6).

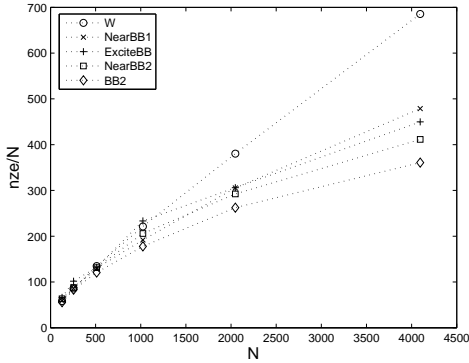
We used the two error measures (9), i.e., the relative solution error and the relative residual error. As a first example, we chose a threshold such that the error for the problem with size $N = 128$ was about 2.0%. The threshold was then scaled with N as suggested by the estimate (22). The results are given in Figure 5. The plotted values are the numbers of



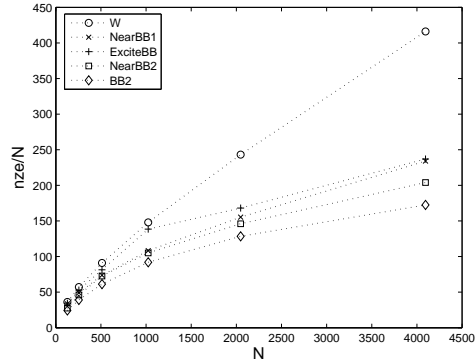
(a) nze/N for the circle, relative solution error



(b) nze/N for the circle, relative residual error



(c) nze/N for the duct, relative solution error



(d) nze/N for the duct, relative residual error

Figure 5: The number of nonzero elements (nze) relative to N after thresholding with a scaling threshold. The initial threshold is such that the error (e_R or e_S) is 2% for $N = 128$.

nonzero elements in the compressed discretization matrix, divided by N . According to the results of [27, 22], we expect the line representing the regular wavelet transformation to be linear, i.e., the number of elements grows quadratically with N . This is clearly visible in the Figure.

The results show a much reduced number of significant elements of the wavelet packet transformations compared to the wavelet transformation. The residual error criterion increases the sparsity level in all cases. The two-dimensional best basis produces the highest sparsity, and is better than the wavelet transformation by a factor of 17 at $N = 4096$ for the circle. The wavelet packet basis captures the inherent problem frequency much better than the classical wavelet basis. The factor for the duct is approximately 2.5. Due to the sharpness of the corners and the complex shape, it appears more difficult for the wavelet packet basis to capture all of the relevant frequencies. The computed errors e_R and e_S are not constant at 2% however, and tend to decrease slowly with N . All values varied between 2.5% and 1% for the

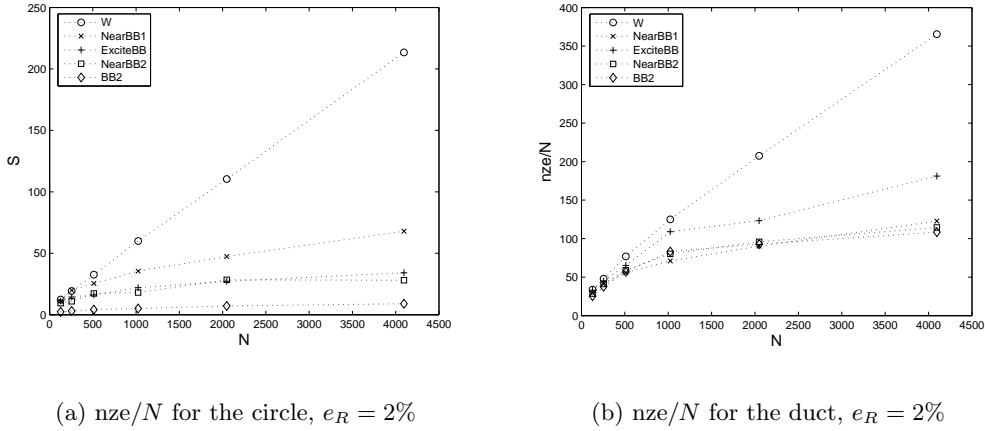


Figure 6: The number of nonzero elements (nze) relative to N after thresholding with an adaptive threshold, such that the relative residual error is fixed at $2 \pm 0.05\%$.

current example. The errors for the wavelet-packet transformations seemed to decrease faster than for the wavelet transformation. These observations indicate that the scaling threshold (22) is somewhat too restrictive, and that the wavelet-packet transformations not only produce more sparsity, but also a more accurate representation. The threshold can therefore be larger than for the wavelet transformation.

To quantify the maximal possible sparsity in this setting, we choose the threshold in the next example such that the error is kept constant at $2.0 \pm 0.05\%$. Of course, in practice this procedure is not feasible, but for analysis purposes it does provide some additional insight. A binary search algorithm, starting with the scaling threshold as initial value, found a suitable threshold value typically in 4 to 6 iterations. The results are given in Figure 6 for the residual error criterion. The threshold values scaled approximately as $O(N^{-1/3})$ for (BB2). The sparsity is much improved: for $N = 4096$, (BB2) for the circle requires 36,596 elements to satisfy the error criterion, compared to 56,842 in the previous example. The difference for the duct is in this case negligible: 468,775 elements for the adaptive threshold, compared to 475,246 for the scaling threshold. The corresponding dense matrix has 16 million elements.

It should be noted that for the duct, (NearBB2) and (BB2) are not significantly better than the one-dimensional (NearBB1). While (BB2) guarantees the best possible compression for a two-dimensional wavelet-packet basis for a fixed threshold, the criterion in which we match the threshold to the final compression error leads to different thresholds for the different methods. The (NearBB2) method is the best one in this case by a small margin. It was verified for other obstacles that the largest gain with (BB2) is obtained for smooth obstacles. For non-smooth obstacles with corners, such as the duct, the top-down approximations (NearBB1) and (NearBB2) perform almost equally well or sometimes even marginally better than (BB2). The other obstacles considered were an ellipse, a rounded gear wheel and an L-shaped domain with corners.

Finally, we would like to examine the asymptotical complexity of the number of nonzero elements in the compressed matrices as a function of N . Assume we can write the number

N_2	scaling threshold					fixed error $e_R = 2\%$				
	256	512	1024	2048	4096	256	512	1024	2048	4096
W	1.74	1.77	1.87	1.91	1.95	1.64	1.76	1.88	1.88	1.95
NearBB1	1.67	1.46	1.46	1.59	1.50	1.86	1.36	1.48	1.41	1.52
RhsBB	1.41	1.37	1.70	1.56	1.45	1.37	1.16	1.45	1.31	1.33
NearBB2	1.19	1.84	1.26	1.79	1.15	1.22	1.66	1.06	1.60	0.98
BB2	1.53	1.44	1.59	1.58	1.51	1.43	1.41	1.28	1.50	1.33

Table 1: Value of β in the estimate $S = O(N^\beta)$ for the circle, using the residual error criterion.

N_2	scaling threshold					fixed error $e_R = 2\%$				
	256	512	1024	2048	4096	256	512	1024	2048	4096
W	1.58	1.61	1.60	1.67	1.75	1.67	1.54	1.60	1.64	1.73
NearBB1	1.37	1.49	1.48	1.50	1.44	1.63	1.29	1.40	1.45	1.26
RhsBB	1.56	1.53	1.50	1.45	1.58	1.77	1.39	1.42	1.36	1.47
NearBB2	1.45	1.45	1.51	1.47	1.42	1.52	1.38	1.38	1.30	1.30
BB2	1.45	1.43	1.50	1.43	1.38	1.54	1.36	1.48	1.37	1.23

Table 2: Value of β in the estimate $S = O(N^\beta)$ for the duct, using the residual error criterion.

of significant elements S as $S = O(N^\beta)$ for some $\beta \in \mathbb{R}$. The value of β can then be estimated from two successive discretizations with N_1 and N_2 unknowns, and S_1 and S_2 significant elements, by $\beta \sim \log(S_1/S_2)/\log(N_1/N_2)$. Tables 1 and 2 show the results for the circle and the duct. The value of β approximates 1.5 for the scaling threshold, but is lower for the threshold that corresponds to a fixed error. The number of elements is therefore empirically $O(N^{1.5})$ for the scaling threshold, and at best $O(N)$ for the computed threshold. A complexity of $O(N^{1.4})$ seems to be a best fit. A rigorous mathematical proof for these empirical complexities has not been found, but an argument that supports the results is given in the next section.

The condition number of the formulation (4) with optimal choice $\alpha = 1/k$ is monotonously but slowly increasing. The values for the test problems are given in Table 3. The condition number for the duct with $N = 4096$ unknowns is 465, compared to the moderately large value of $2.5e4$ for the formulation with the common choice of $\alpha = 1$. The latter formulation leads to more irregular behaviour for the condition number. It can be seen from formula (6) for the matrix elements, that the contribution of $\frac{I}{2} + D^*$ to the discretization dominates for large values of k . The ill conditioning is therefore still caused by the resonant eigenvalues. The choice $\alpha = 1/k$ completely eliminates this behaviour.

6 Computational complexity of the matrix-vector product

6.1 Sparsity of the discretization matrix

The numerical results demonstrate the improved sparsity of the discretization matrix when using wavelet packets as basis functions. A rigorous proof of the reduced complexity would be rather involved, due to the adaptive nature of the algorithm. However, an estimate is

N	$\alpha = 1$						$\alpha = 1/k$					
	256	512	1024	2048	4096	8192	256	512	1024	2048	4096	8192
circle	101.6	20.0	92.0	852	197	3.7e3	4.3	5.4	6.7	8.5	10.7	13.4
duct	1.7e2	1.1e3	6.1e2	1.1e3	2.5e4	1.1e4	26.8	27.2	40.4	87.5	465	448

Table 3: Condition number of the discretization matrix for different choices of the coupling parameter α in the Combined Field Integral Equation.

derived below. In addition, insights are gained into the behaviour of the method.

Denote by $\hat{w}_n(\xi)$ the Fourier transform of the wavelet packet function $w_{n,0,0}(x)$. Since $w_{n,0,0}(x)$ is compactly supported, the same can not hold for $\hat{w}_n(\xi)$. However, the main energy of $\hat{w}_n(\xi)$ is located inside a certain frequency band that depends on n . Its size can be estimated from the variance

$$\sigma_n := \inf_{\xi_0 \in \mathbb{R}} \int_0^\infty |\xi - \xi_0|^2 |\hat{w}_n(\xi)|^2 d\xi.$$

Ideally, each wavelet packet basis function $w_{n,0,0}(x)$ has a frequency spectrum inside a band of fixed size, that does not overlap with the spectrum of other basis functions. This is only approximately the case. It is shown that $\sigma_n \sim n^\delta$ with $\delta > 0$ a small constant [9]. The size of the band is approximately proportional to the standard deviation

$$s_n := \sqrt{\sigma_n} = n^{\delta/2}. \quad (23)$$

Consider a function that is sampled at $N = 2^L$ equispaced points. The entire frequency spectrum is given by $0 \leq f < 2^L$. In the ideal case, $w_{n,j,k}(x)$ has a frequency spectrum of the form $f \in [2^j(\xi_0 - 1/2), 2^j(\xi_0 + 1/2)]$, i.e., a band of fixed width 2^j , with ξ_0 the average frequency of $\hat{w}_n(x)$. The basis functions $w_{n,j,k}(x)$, $n = 0, \dots, 2^{L-j} - 1$, then cover the entire spectrum independently from each other. Now consider the function $\cos(2\pi fx)$, with a frequency f that is proportional to N . A value of ξ_0 , and a corresponding value of n , can be found for any fixed scale j such that $f \in [2^j(\xi_0 - 1/2), 2^j(\xi_0 + 1/2)]$. Hence, the cosine can be represented accurately on scale j by only 2^j basis functions $w_{n,j,k}(x)$, $k = 0, \dots, 2^j - 1$, independently of N . Both ξ_0 and n scale linearly with N .

Now assume a bandwidth of $n^{\delta/2}$. Then there are $O(n^{\delta/2})$ intervals of the form $[2^j(\xi_0 - O(n^{\delta/2}), 2^j(\xi_0 + O(n^{\delta/2}))]$ that contain f . Hence, the total number of coefficients required to represent the cosine accurately on scale j has order $O(2^j n^{\delta/2}) = O(2^j N^{\delta/2})$. The value of δ can be determined easily for a given wavelet family from a numerical experiment. We have computed the number of coefficients larger than a fixed threshold for the function $\cos(2\pi fx)$, $x \in [0, 1]$, with $N = 10f$. The result is shown for a wavelet approximation and a best basis approximation in Figure 7. The number of coefficients in the wavelet approximation scales linearly in N . For the best basis approximation, we have $\delta/2 \approx 0.7$.

In two dimensions, the number of wavelet coefficients scales as $O(N^2)$, and the number of coefficients in the best basis approximation scales as $O(N^{2\delta/2}) = O(N^{1.36})$. This estimate agrees reasonably well with the results of the previous section. The simplified model, i.e., the compression of a two-dimensional function with an inherent frequency, illustrates the principles of the compression of the discretization matrix. The influence of the singularity may be discarded in this estimate, as it can be represented locally on each scale with only few coefficients.

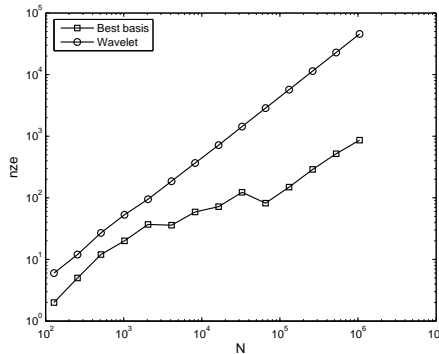


Figure 7: The number of nonzero elements after compression of $\cos(2\pi fx)$ with $N = 10f$.

6.2 Comparison to existing methods

Various methods have been proposed in recent years for the solution of wave problems at high frequencies by boundary element methods. The Fast Multipole Method has been extended for high frequency problems, yielding a computational complexity of $O(N \log N)$ in [12]. However, the constants involved are reported to be large. The same complexity can be obtained, again with large constants reported, when using hierarchical basis functions with Hierarchical Matrices [3]. Once the compressed discretization matrix in our approach is computed, the number of operations in a matrix-vector product is essentially proportional to the number of significant elements, with a low proportionality constant.

A technique that has perhaps more in common with our described approach, is the use of plane waves as basis functions[26]. Similarly to suitably chosen wavelet packets, plane wave basis functions incorporate the wave nature of the problem. Numerical results indicate that the typical requirement of 10 basis functions per wavelength is reduced to 2–4 basis functions. This leads to very significant savings. However, this approach only reduces the constants, and does not lower the order of the computational complexity of a matrix-vector product.

Finally, in some cases, the solution of the integral equation may be modelled by a smooth function times a very oscillatory, but known, ansatz function. A discretization approach can then be devised to approximate the non-oscillating smooth function, with a number of operations that is actually independent of the wavenumber [6]. However, currently, this approach is limited to convex obstacles.

7 Concluding remarks

The two-dimensional wavelet packet transformation is suitable for the compression of the discretization matrix of an oscillatory integral equation. Results previously reported were improved with a factor of up to 6 by the use of this transformation for problems with a smooth boundary. The computational complexity was numerically shown to be approximately $O(N^{1.4})$, which should be compared to $O(N^2)$ for the regular wavelet transformation. The number of matrix-vector products was minimized by the chosen integral equation formulation of the problem. Further, due to a new formulation of the wavelet-packet transformed matrix-

vector product, they can be computed by existing software that is optimized for this purpose. The cost of the setup phase is increased. The method is particularly effective if a solution is required for multiple boundary conditions, or if the solution phase dominates the total computation time. The cost of the wavelet packet transformation can be reduced by approximating the best basis algorithm with a top-down approach. The sparsity is then reduced by a factor of approximately 2 for smooth boundaries, while the results are equally good for boundaries with corners.

References

- [1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover Publications, New York, 1965.
- [2] S. Amini. On the choice of the coupling parameter in boundary integral formulations of the exterior acoustic problem. *Appl. Anal.*, 35:75–92, 1990.
- [3] L. Banjai and W. Hackbusch. H - and H^2 -matrices for low and high frequency helmholtz equation. Technical Report 17, Max-Planck-Institut für Mathematik in den Naturwissenschaften, 2005.
- [4] G. Beylkin, R. R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms I. *Comm. Pure Appl. Math.*, 44:141–183, 1991.
- [5] S. Börm and W. Hackbusch. Data-sparse approximation by adaptive H^2 -matrices. *Computing*, 69:1–35, 2002.
- [6] O. Bruno, C. Geuzaine, J. Monro, and F. Reitich. Prescribed error tolerances within fixed computational times for scattering problems of arbitrarily high frequency: the convex case. *Phil. Trans. R. Soc. Lond. A.*, 362(1816):629–645, 2004.
- [7] A. Cohen and I. Daubechies. On the instability of arbitrary biorthogonal wavelet packets. *SIAM J. Math. Analysis*, 24:1340–1354, 1993.
- [8] R. R. Coifman, Y. Meyer, S. R. Quake, and M. V. Wickerhauser. Signal processing and compression with wavelet packets. In Y. Meyer and S. Roques, editors, *Progress in Wavelet Analysis and Applications*, pages 77–93. Editions Frontières, 1993.
- [9] R. R. Coifman, Y. Meyer, and M. V. Wickerhauser. Size properties of wavelet packets. In M. B. Ruskai, G. Beylkin, R. R. Coifman, I. Daubechies, S. Mallat, Y. Meyer, and L. Raphael, editors, *Wavelets and Their Applications*, pages 453–470. Jones and Bartlett, Boston, 1992.
- [10] R. R. Coifman and M. V. Wickerhauser. Entropy based algorithms for best basis selection. *IEEE Trans. on Information Theory*, 32:712–718, March 1992.
- [11] W. Dahmen, H. Harbrecht, and R. Schneider. Compression techniques for boundary integral equations – optimal complexity estimates. IGPM report 02-06, RWTH Aachen, 2002.

- [12] E. Darve. The fast multipole method: Numerical implementation. *J. Comput. Phys.*, 160:195–240, 2000.
- [13] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.*, 41:909–996, 1988.
- [14] I. Daubechies. *Ten lectures on wavelets*. SIAM, Philadelphia, PA, 1992.
- [15] H. Deng and H. Ling. Fast solution of electromagnetic integral equations using adaptive wavelet packet transform. *IEEE Trans. Antennas and Propag.*, 47(4):674–682, 1999.
- [16] H. Deng and H. Ling. On a class of predefined wavelet packet bases for efficient representation of electromagnetic integral equations. *IEEE Trans. Antennas and Propag.*, 47(12):1772–1779, dec 1999.
- [17] W. L. Golik. Wavelet packets for fast solution of electromagnetic integral equations. *IEEE Trans. Antennas and Propag.*, 46(5):618–624, may 1998.
- [18] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73(2):325–348, 1987.
- [19] W. Hackbusch and Z. Novak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numer. Math*, 54:463–491, 1989.
- [20] R. F. Harrington. *Field Computation by Moment Methods*. Macmillan, New-York, 1968.
- [21] N. Hess-Nielsen and M. V. Wickerhauser. Wavelets and time-frequency analysis. *Proceedings of the IEEE*, 84(4):523–540, 1996.
- [22] D. Huybrechs, J. Simoens, and S. Vandewalle. A note on wave number dependence of wavelet matrix compression for integral equations with oscillatory kernel. *J. Comput. Appl. Math.*, 172(2):233–246, 2004.
- [23] D. Huybrechs and S. Vandewalle. Quadrature formulae for wavelet approximations of piecewise smooth or singular functions. *J. Comput. Appl. Math.*, 180(1):119–135, 2004.
- [24] W. McLean. *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge University Press, 2000.
- [25] J.-C. Nédélec. *Acoustic and Electromagnetic Equations*, volume 144 of *Applied Mathematical Sciences*. Springer, 2001.
- [26] E. Perrey-Debain, J. Trevelyan, and P. Bettess. Plane wave interpolation in direct collocation boundary element method for radiation and wave scattering: numerical aspects and applications. *J. Sound and Vib.*, 261(5):839–858, 2003.
- [27] R. L. Wagner and W. C. Chew. A study of wavelets for the solution of electromagnetic integral equations. *IEEE Trans. Antennas and Propag.*, 43(8):802–810, 1995.
- [28] M. V. Wickerhauser. Nonstandard matrix multiplication. Preprint, Yale University, May 1990.