Numer. Math. Theor. Meth. Appl.
doi: 10.4208/nmtma.OA-2022-0083

Vol. **16**, No. 1, pp. 26-57
February 2023

# Application of Adapted-Bubbles to the Helmholtz Equation with Large Wavenumbers in 2D

Adem Kaya*

*Institut für Mathematik, Universität Potsdam, Karl-Liebknecht-Str., 24-25 14476 Potsdam/Golm, Germany*

Received 17 May 2022; Accepted (in revised version) 31 July 2022

**Abstract.** An adapted-bubbles approach which is a modification of the residual-free bubbles (RFB) method, is proposed for the Helmholtz problem in 2D. A new two-level finite element method is introduced for the approximations of the bubble functions. Unlike the other equations such as the advection-diffusion equation, RFB method when applied to the Helmholtz equation, does not depend on another stabilized method to obtain approximations to the solutions of the sub-problems. Adapted-bubbles (AB) are obtained by a simple modification of the sub-problems. This modification increases the accuracy of the numerical solution impressively. We provide numerical experiments with the AB method up to $ch = 5$ where $c$ is the wavenumber and $h$ is the mesh size. Numerical tests show that the AB method is better by far than higher order methods available in the literature.

**AMS subject classifications**: 65N30, 65N06

**Key words**: Helmholtz equation, adapted-bubbles, residual-free bubbles, two-level finite element.

## 1. Introduction

Enriching linear finite element space with residual-free bubble functions is a general framework for the discretizations of the partial differential equations such as the Helmholtz equation [12], advection-diffusion equation [3,14] and Navier-Stokes equation [20]. These functions strongly satisfy the original differential equations and hence obtaining the bubble functions is generally as difficult as solving the original problem such as the advection-diffusion equation [3]. Unlike it was stated in [13], we will show that this is not the case for the Helmholtz problem. Obtaining the bubble functions on triangular elements is easier than solving the original problem. The standard Galerkin finite element method can be used with a coarse mesh to obtain efficient approximations to the bubble functions.

*Corresponding author. *Email address:* baffiros@gmail.com (A. Kaya)

 

The residual-free bubbles method produces the exact solution of linear differential equations in the one-dimensional case. However, the method in higher-dimensions is approximate and as we will show for the Helmholtz problem in this article, its contribution to the stabilization of the standard Galerkin method is not well adjusted. We modify the residual-free bubbles (RFB) method in 2D by multiplying the right-hand side of the bubble equations with a constant. This operation impressively increases the accuracy. The new bubbles are no more residual-free and we call them adapted-bubbles (AB). We provide the optimal values of the constants for the triangular and rectangular elements separately. We apply a two-level finite element method using the standard linear Galerkin finite element method to get approximations to the bubble functions.

We provide analysis to show how the AB method mitigates the pollution error. To this end, we approximate the bubble functions with piece-wise defined linear functions so-called pseudo-bubbles. The analysis give rise to a fourth-order finite difference scheme with seven-point stencil for the plane waves. It is perfectly applicable on polygonal and triangular domains. We use this method to do comparison with the AB method.

Standard discretizations when applied to the Helmholtz problem suffer from the pollution effect when the wavenumber is large [2]. Moreover standard iterative solvers are ineffective in obtaining the solutions of the discrete Helmholtz equation [9]. There is a great effort in literature to overcome these difficulties. Among the discretization techniques, there are finite difference [10,24], finite element [1,18,27], discontinuous Galerkin [6, 11], virtual element [21], and boundary element methods [19]. At the same time, there is a great effort to develop efficient preconditioners, such as multigrid [4,7,8,15,26] and domain decomposition methods [16,17,25].

The AB method proposed in this article works for very large $ch$. We provide numerical experiments up to $ch = 5$ but, in principle, it works for larger $ch$ as long as the parameters are obtained with the Algorithm 1 proposed in this article. Numerical results show that the AB method is better by far than higher order methods available in the literature. It works in the regime ($ch > 3.5$) where the other methods do not work.

The rest of this paper is organized as follows. We review the RFB method for the Helmholtz equation in Section 2. We explain how to implement two-level finite element method in 1D and provide analysis to show the contribution of the bubble functions in reducing the pollution error in Section 3. Section 4 is devoted to the analysis of the RFB method in 2D. We propose the AB method for triangular elements in Section 5. The AB method is considered with rectangular elements in Section 6. We finish with concluding remarks in Section 7.

## 2. Residual-free bubbles method (RFB) for the Helmholtz equation

We start with considering the Helmholtz problem in 1D with Dirichlet boundary conditions on unit interval

$$\begin{cases} \mathcal{L}u = -u''(x) - c^2 u(x) = f(x), \quad x \in I = (0,1), \\ u(0) = 0, \quad u(1) = \sin(c), \end{cases} \tag{2.1}$$

where we assume that the wavenumber $c$ is constant and $c^2$ is not an eigenvalue of the Dirichlet-Laplacian so that the problem has a unique solution. RFB method can be summarized as follows. Let us start with recalling abstract variational formulation of (2.1): Find $u \in H^1(I)$ such that

$$a(u,v) = (f,v), \quad u(0) = 0, \quad u(1) = \sin(c), \quad \forall v \in H_0^1(I),$$

where

$$a(u,v) = \int_I u'v'dx - c^2 \int_I uv\,dx,$$
$$(f,v) = \int_I fv\,dx.$$

Define $V_h \subset H_0^1(I)$ (can be a complex-valued function space) as a finite-dimensional space. Let $x_i$, $i = 1,\ldots,n$ be the uniformly distributed nodes obtained by the decomposition of the domain $I$ such that the mesh size $h = 1/(n-1)$. Then the Galerkin finite element method reads: Find $u_h \in V_h$ such that

$$a(u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h.$$

We now decompose the space $V_h$ as $V_h = V_L \bigoplus V_B$, where $V_L$ is the space of continuous piecewise linear polynomials defined on the nodes $x_i$ and $V_B = \bigoplus_K B_K$ with $B_K = H_0^1(K)$. From this decomposition, every $v_h \in V_h$ can be written in the form $v_h = v_L + v_B$, where $v_L \in V_L$ and $v_B \in V_B$. Bubble component $u_B$ of $u_h$ satisfies the original differential equation in an element $K$ strongly, i.e.

$$\mathcal{L}u_B = -\mathcal{L}u_L + f \quad \text{in } K, \tag{2.2}$$

subject to boundary condition

$$u_B = 0 \quad \text{on } \partial K. \tag{2.3}$$

Since the support of the bubble $u_B$ is contained within the element $K$, we can make a static condensation for the bubble part, getting directly the $V_L$- projection $u_L$ of the solution $u_h$ [12]. This can be done as follows. Using $V_h = V_L \bigoplus V_B$, the finite element approximation reads: Find $u_h = u_L + u_B$ in $V_h$ such that

$$a(u_L, v_L) + a(u_B, v_L) = (f, v_L), \quad \forall v_L \in V_L. \tag{2.4}$$

## 3. A two level finite element method

In order to find bubble part $u_B$ of the solution, we need to solve (2.2)-(2.3). The problem defined by Eqs. (2.2)-(2.3) is addressed by solving instead

$$\begin{cases} -\varphi_i'' - c^2\varphi_i = c^2\psi_i & \text{in } K, \quad i = 1,\ldots,n_{en}, \\ \varphi_i = 0 & \text{on } \partial K, \end{cases} \tag{3.1}$$

and

$$\begin{cases} -\varphi_f'' - c^2\varphi_f = f & \text{in } K, \\ \varphi_f = 0 & \text{on } \partial K, \end{cases} \tag{3.2}$$

where $n_{en} = 2$ is the number of element nodes, $\psi_i$ are global basis functions restricted to element $K$ and $\varphi_i$ are basis functions. Thus if

$$u_L|_K = \sum_i d_i^K \psi_i, \tag{3.3}$$

then

$$u_B|_K = \sum_i d_i^K \varphi_i + \varphi_f, \tag{3.4}$$

where $d_i^K$ are constant coefficients. It is clear that applying the operator $\mathcal{L}$ to (3.4), we recover (2.2)

$$\mathcal{L}u_B|_K = \sum_i d_i^K \mathcal{L}\varphi_i + \mathcal{L}\varphi_f = c^2 \sum_i d_i^K \psi_i + f = -\mathcal{L}u_L|_K + f.$$

Substituting (3.3) and (3.4) into (2.4), we get the matrix formulation

$$\sum_K \sum_i^{n_{en}} d_i^K \big((\psi_i', \psi_j') - c^2(\psi_i, \psi_j) - c^2(\varphi_i, \psi_j)\big) = (f, \psi_j) \tag{3.5}$$

at the global level where $d_i^K$ are the finite element approximations to the solutions at the nodes.

Numerical solution of the bubble problems (3.1) and (3.2) generally requires using a nonstandard method such as for the case of the advection-diffusion equation. This makes the RFB method dependent on another stabilized method when applying it as a two-level finite element method. In [13], the Galerkin-least-squares method (GLS) was used to get approximations to the bubble functions in solving the Helmholtz equation. Although this is true for the advection-diffusion equation, there is no need to use a nonstandard method to get approximations to the bubble functions when the Helmholtz problem is under consideration. We explain this fact in 1D. Suppose that we have a discretization of the domain such that $ch = 0.6$. Even if we use $3$ nodes on the sub-domain (element), $ch_e$ ($h_e$ is mesh size on the sub-domain) becomes less than $0.6$. If $ch = 3$ on the global mesh, then using $11$ nodes for the sub-problems makes $ch_e = 0.3$. More precisely, it is always true that $ch_e < ch$.

It is true that GLS computation is known to incur at most marginal increase in computational cost over the standard Galerkin method. However, GLS for the sub-problems may lead to misinterpretations related to the bubble functions.

Another way of obtaining the bubble function is to use separation of variables when rectangular elements are used [12]. However, this gives rise to a series solution of the bubble function for which it must be truncated. For a good accuracy, $200$ terms are used in [12] which is computationally not so effective. Another drawback is that this approach is limited to the rectangular elements.

### 3.1. Analysis of the pollution effect of the sub-problems

It is always true that $ch_e < ch$, where $h_e$ and $h$ are mesh sizes of the sub-problem and global problem, respectively. This is an indication that the sub-problems are easier to solve; however, we must analyse the pollution effect for the sub-problems for large wavenumbers. It is well known that the difficulty in solving the Helmholtz problem occurs when the exact solution is very oscillatory [2]. In our case, the exact solutions of the sub-problems are not oscillatory. When $ch < \pi$, the exact solutions of the sub-problems are always in the form of a half wave as the homogenous Dirichlet boundary condition is applied everywhere on the boundary. When $\pi < ch < 2\pi$, the exact solutions of the sub-problems are always in the form of a single wave. In this regime, the standard Galerkin method is pollution free for the sub-problems for any wavenumber. Note that, 10-12 nodes ($G$) per wave is generally chosen by engineers for which $ch \in [0.5, 0.65]$ from the relation $G = 2\pi/(ch)$ [5].

We use standard Galerkin finite element method with piecewise linear basis functions to approximate the bubble functions. Note that the bubble problems (3.1) and (3.2) can be solved independently and hence parallel processors can be used to carry out these computations efficiently. When uniform meshes are used and the right-hand side function $f$ is constant, construction of the system matrix is as cheap as construction of the system matrix of the standard Galerkin finite element method.

### 3.2. Shape of the bubble functions and pseudo-bubbles

We have shown that the RFB method is not dependent on another stabilized method to get approximations to the bubble functions when a two-level finite element method is used due to the non-oscillatory behavior of the exact solutions of the sub-problems. This non-oscillatory behavior of the exact solutions opens a gateway to approximate these bubble functions with piecewise-defined linear simple functions. These approximations are called pseudo-bubbles and constructed considering the shape of the bubble functions. Pseudo-bubbles were applied to the advection-diffusion-reaction equation in [22, 23].

Here, we consider the case $ch < \pi$ for which the bubble functions are in the form of a half wave. We present the bubble functions $\varphi_{1,2}$ in Fig. 1 for $c = 60, 300$ when $h = 0.01$. Efficient yet cheap approximations to these bubble functions with piecewise-defined linear functions are given in Fig. 2. While on the left in Fig. 2, two pseudo-bubbles are used, it is possible to approximate $\varphi_{1,2}$ with a single pseudo-bubble. The advantage of using a single pseudo-bubble is that the maximum of the pseudo-bubble occurs in the middle of the element. Applying the minimization technique applied in [22], one can find the optimal heights and locations of the peaks of the pseudo-bubbles. When a single pseudo-bubble is used, one can easily calculate integrals in the finite element formulation. This will be important in modifying the RFB method in 2D.

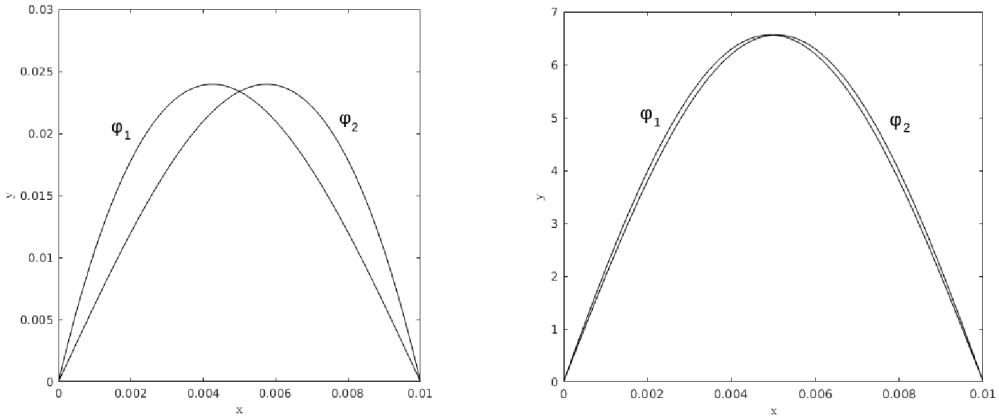


Figure 1: Bubble functions for $c = 60$ (left) and $c = 300$ (right).

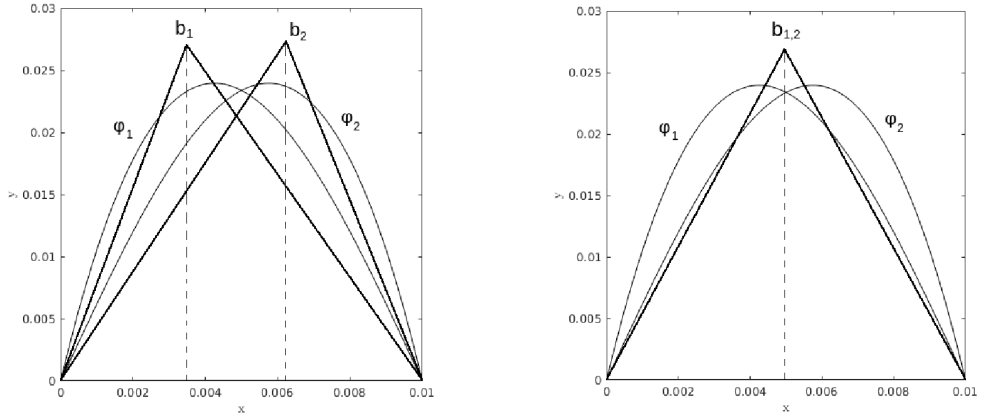


Figure 2: Two different approaches to approximate the bubble functions.

### 3.3. Analysis of the pseudo-bubbles

In order to see how the residual-free bubbles method overcome the pollution effect, we first consider linear finite element method for (2.1) when $f(x) = 0$, $u(0) = 0$ and $u(1) = \sin(c)$. The exact solution is $u(x) = \sin(cx)$. We obtain the truncation error and see how the error deteriorates as $c$ increases for fixed $ch$ which is simply the pollution effect. To this end, we jump to the finite difference equivalence of the linear finite element method. Let $U_j$ represents the numerical solution and choose $n$ equally distributed nodes for which $h = 1/(n-1)$. Taking the integrals in linear finite element formulation and scaling by $h$ gives

$$-\frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} - c^2 \frac{U_{j+1} + 4U_j + U_{j-1}}{6} = 0, \quad j = 2, \ldots, n-2. \tag{3.6}$$

From the boundary conditions

$$U_1 = 0, \quad U_n = \sin(c).$$

Using the Taylor series expansion, we get the truncation error for (3.6)

$$\tau(x) = -\frac{c^2h^2}{6}u'' - \left(\frac{c^2h^4}{72} + \frac{h^2}{12}\right)u^{(4)} - \frac{h^4}{360}u^{(6)} + \mathcal{O}(h^6). \tag{3.7}$$

The pollution effect can not be seen from (3.7). To see it, we substitute the exact solution $u(x) = \sin(cx)$ into (3.7)

$$\tau(x) = \frac{c^4h^2}{6}\sin(cx) - \left(\frac{c^6h^4}{72} + \frac{c^4h^2}{12}\right)\sin(cx) + \frac{c^6h^4}{360}\sin(cx) + \mathcal{O}(c^8h^6).$$

Rearranging the above equation gives

$$\tau(x) = \sin(cx)\left(\frac{c^4h^2}{12} - \frac{c^6h^4}{90} + \mathcal{O}(c^8h^6)\right). \tag{3.8}$$

When the exact solution is oscillatory, that is, $c$ is large, the term $c^4h^2/12$ in (3.8) becomes large, that is, $\tau(x)$ is large, even if $ch =$ constant is small. This is called the pollution effect.

The simplest way to mitigate this pollution effect is to choose $c^2h$ sufficiently small. However, this requires intractable matrices in higher dimensions. The general idea in literature is to decrease the effect of the first few terms in (3.8) so that tractable matrix sizes can be obtained. For example, using higher order accurate methods of finite difference or higher order polynomials finite element may allow to eliminate the first few terms. If the first term can be eliminated, then the requirement to mitigate the pollution effect reduces to $c^{3/2}h$ being sufficiently small. However, higher order methods generally use more points and this increases the nonzero entries of the corresponding matrix.

In order to get a deeper insight of the working principle of the residual-free bubbles method in mitigating the pollution effect, we consider the pseudo-bubbles in Fig. 2 on the right. This choice allows us to take the integrals containing the bubble functions, explicitly.

We can define the pseudo-bubbles $b_{1,2}$ using the basis functions $b_{1,2}^T$ represented in Fig. 3 and heights of $b_{1,2}$, i.e., $\alpha_{1,2}$. More precisely,

$$b_{1,2} = \alpha_{1,2}b_{1,2}^T, \tag{3.9}$$

where $\alpha_1 = \alpha_2$. Applying the technique proposed in [22] (set $\xi = h/2, \epsilon = 1, \sigma = -c^2$ in [22, Eq. 13]) gives

$$\alpha_1 = \frac{3c^2h^2}{4(12 - c^2h^2)}. \tag{3.10}$$

Taking the integrals in (2.4) and making use of (3.9) give the finite difference formula

$$-\frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} - c^2\frac{U_{j+1} + 4U_j + U_{j-1}}{6} - \alpha_1 c^2\frac{U_{j+1} + 2U_j + U_{j-1}}{4} = 0,$$
$$j = 2, \ldots, n-2. \tag{3.11}$$

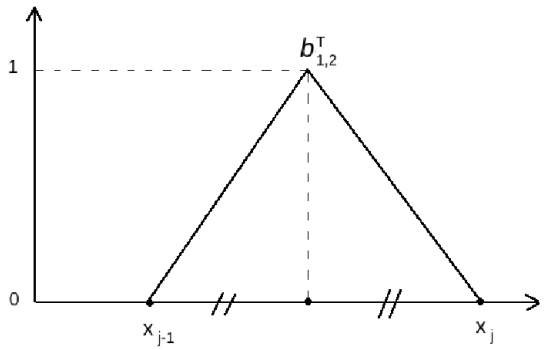


Figure 3: Basis functions employed in the approximation of bubble functions.

Using Taylor expansions of $u(x \pm h)$, definition of $\alpha_1$ given in (3.10) and the exact solution $u(x) = \sin(cx)$, we obtain the truncation error

$$\tau(x) = \sin(cx)\left(\frac{c^4h^2}{12} - \frac{c^6h^4}{90} + \mathcal{O}(c^8h^6)\right) + \sin(cx)\left(-\frac{3c^4h^2}{4(12-c^2h^2)} + \frac{3c^6h^4}{16(12-c^2h^2)} + \mathcal{O}(c^8h^6)\right).$$

Rearranging the right-hand side of the above formula we end up with

$$\tau(x) = \sin(cx)\left(c^4h^2\left(\frac{1}{12} - \frac{3}{4(12-c^2h^2)}\right) + c^6h^4\left(-\frac{1}{90} + \frac{3}{16(12-c^2h^2)}\right) + \mathcal{O}(c^8h^6)\right). \tag{3.12}$$

As $h \to 0$, $\tau(x)$ behaves like

$$\sin(cx)\left(\frac{c^4h^2}{48} - \frac{13}{2880}c^6h^4 + \mathcal{O}(c^8h^6)\right). \tag{3.13}$$

Comparing (3.8) with (3.13), we see that the coefficients of $c^4h^2$ and $c^6h^4$ are decreased in magnitude. The approximate bubbles shows how the pollution effect is reduced. It is known that the RFB method for 1D linear equations is exact [13]. This means that it automatically makes the coefficient of all powers $c^{n+2}h^n$, $n = 2, 3, \ldots,$ zero. A good approximation to the residual-free bubbles significantly reduce the pollution effect. The RFB method is approximate in 2D. The observations we made here will be helpful to further increase the accuracy of the method in 2D. We will modify the sub-problems in 2D and use adapted-bubbles to further increase the accuracy of the bubble approach.

### 3.4. Effect of the interpolation error of the sub-problems

It is shown that solution of the sub-problems can be approximated by standard linear Galerkin finite element method. However, it is important to see the effect of the interpolation error that occurs after solving the sub-problems. In 1D, the sub-problems are exactly solvable. Using the exact solutions, we construct linear interpolations using $N_s$ nodes on each global mesh and measure the error in infinity norm. In Fig. 4, we report the error for different values of $ch$ and $N_s$ for varying $c$ for the problem in (2.1). It is clear that, when $ch = 0.6$, the interpolation error does not much effect the global error. However, for larger $ch$, the global error deteriorates as $c$ increases. This suggests that, one should increase the number of elements used for the sub-problems for larger $ch$ to keep the global error under control. However, this requires solving large linear system of equations in higher dimensions. One of the way to compensate for the interpolation error is to multiplying the right-hand side of the bubble problems (3.1) with a constant, say $\mu$. The problem of finding optimal values of $\mu$ is explained in 2D (Section 5) in combination with handling another problem that is specific to 2D.

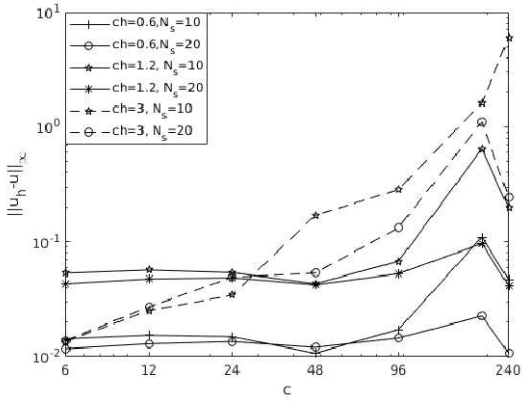


Figure 4: Effect of the interpolation error of the sub-problems in the global error.

## 4. The RFB method in 2D

We have shown that the RFB method is able to solve the Helmholtz problem in 1D cheaply and efficiently for very large wavenumbers. As it was stated in [13], RFB method is not as efficient in 2D as in 1D. To show this fact, we consider the following problem on an equilateral triangular shaped domain $\Omega$ with vertices $(0,0)$, $(0,1)$ and $(0.5, \sqrt{3}/2)$:

$$\begin{cases} -\Delta u - c^2 u = 0 & \text{in } \Omega, \\ u(x,y) = \sin\left(cy\sin(\theta) + cx\cos(\theta)\right) & \text{on } \partial\Omega_D, \end{cases} \tag{4.1}$$

where the exact solution is $u(x,y) = \sin(cy\sin(\theta) + cx\cos(\theta))$. We use equilateral triangular elements with linear basis functions. We solve the following equations on

element level to get the RFB functions:

$$\begin{cases} -\Delta\varphi_i - c^2\varphi_i = c^2\psi_i & \text{in } K, \quad i = 1, \dots, n_{en}, \\ \varphi_i = 0 & \text{on } \partial K, \end{cases} \tag{4.2}$$

and

$$\begin{cases} -\Delta\varphi_f - c^2\varphi_f = f & \text{in } K, \\ \varphi_f = 0 & \text{on } \partial K, \end{cases} \tag{4.3}$$

where $n_{en} = 3$, $\varphi_i$ and $\psi_i$, $(i = 1, 2, 3)$ are the RFB and the linear basis functions, respectively. Linear finite element method with a coarse mesh can be used to obtain efficient approximations to the bubble functions. To do some analyses, we approximate the RFB functions with piecewise-defined linear functions with the maximum at the centroid of the element. Let $b_{1,2,3} = \alpha_2 b^T$ be the approximation to the bubble functions where $b^T$ is the linear basis bubble function that assumes zero at the vertices of the element and one at the centroid of the element. Applying the same procedure we applied in 1D (see [23] for more details), gives

$$\alpha_2^2 \int_K \nabla b^T \nabla b^T dS - \alpha_2^2 c^2 \int_K b^T b^T dS = \alpha_2 \int_K \psi_i b^T dS, \quad i = 1, 2, 3. \tag{4.4}$$

Solving the above equation for $\alpha_2$ and calculating the integrals for $\psi_1$ gives

$$\alpha_2 = \frac{2c^2h^2}{3(72 - c^2h^2)}. \tag{4.5}$$

Considering 6 adjacent elements as shown in Fig. 5, the RFB method is equivalent to the following finite difference scheme:

$$\begin{aligned} &\frac{1}{\sqrt{3}h^2}\big(6U_{i,j} - U_{i-1,j} - U_{i-1,j+1} - U_{i+1,j+1} - U_{i+1,j} - U_{i+1,j-1} - U_{i-1,j-1}\big) \\ &\quad - \frac{c^2}{8\sqrt{3}}\big(6U_{i,j} + U_{i-1,j} + U_{i-1,j+1} + U_{i+1,j+1} + U_{i+1,j} + U_{i+1,j-1} + U_{i-1,j-1}\big) \\ &\quad - \frac{\alpha_2 c^2}{6\sqrt{3}}\big(3U_{i,j} + U_{i-1,j} + U_{i-1,j+1} + U_{i+1,j+1} + U_{i+1,j} + U_{i+1,j-1} + U_{i-1,j-1}\big) = 0, \\ &\quad i, j = 1, \dots, N_{\text{int}}, \end{aligned} \tag{4.6}$$

where

$$\begin{aligned} &U_{i,j} \approx u(x, y), \quad U_{i-1,j} \approx u(x - h, y), \quad U_{i-1,j+1} \approx u\Big(x - \frac{h}{2}, y + \frac{\sqrt{3}}{2}h\Big), \\ &U_{i+1,j+1} \approx u\Big(x + \frac{h}{2}, y + \frac{\sqrt{3}}{2}h\Big), \quad U_{i+1,j} \approx u(x + h, y), \\ &U_{i+1,j-1} \approx u\Big(x + \frac{h}{2}, y - \frac{\sqrt{3}}{2}h\Big), \quad U_{i-1,j-1} \approx u\Big(x - \frac{h}{2}, y - \frac{\sqrt{3}}{2}h\Big). \end{aligned}$$

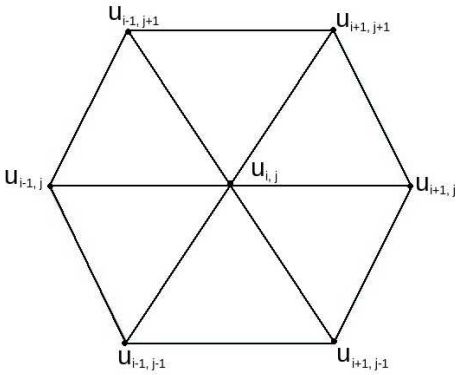


Figure 5: 7 point stencil.

Note that when $\alpha_2 = 0$, (4.6) is equivalent to the linear finite element method with equilateral triangular element.

To analyse the RFB method, we substitute the Taylor expansions of the exact solution at the grid points. The derivation of the truncation error is given in (4.7)-(4.9). In our analysis, we will examine the coefficients of $c^4h^2$ and $c^6h^4$ in (4.9), that is, $C_2$ and $C_1$ in (4.10). Fig. 6 shows the graph of $C_1$ and $C_2$ for $\alpha_2 = 0$ (standard Galerkin) and for $\alpha_2$ in (4.5) (pseudo-RFB). We set $\theta = \pi/3$ to plot the graph of $C_2$. The slight decreases in $C_1$ and $C_2$ in magnitude for $0 < ch < 3$, explains why the RFB method is not effective in 2D

$$\tau(x,y) = \frac{1}{\sqrt{3}h^2}\Bigg(6u(x,y) - u(x-h,y) - u\Big(x-\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) - u\Big(x+\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) - u(x+h,y) - u\Big(x+\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big) - u\Big(x-\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big)\Bigg)$$

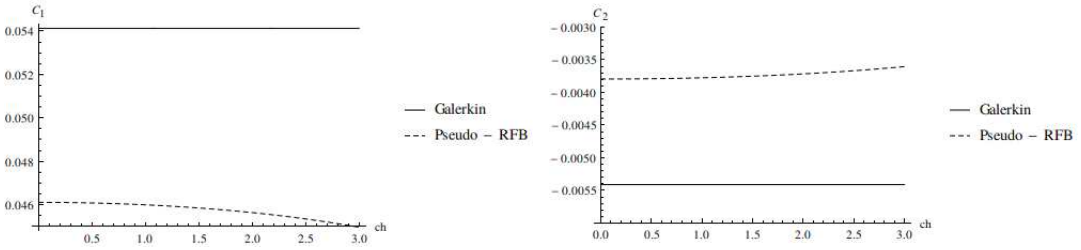


Figure 6: Comparison of the linear Galerkin method and the pseudo-RFB method for the coefficients $C_1$ (left) and $C_2$ (right) when $\theta = \pi/3$.

$$
\begin{aligned}
&- \frac{c^2}{8\sqrt{3}}\Bigg(6u(x,y) + u(x-h,y) + u\Big(x-\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) \\
&\qquad + u\Big(x+\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) + u(x+h,y) \\
&\qquad + u\Big(x+\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big) + u\Big(x-\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big)\Bigg) \\
&- \frac{c^2\alpha_2}{6\sqrt{3}}\Bigg(3u(x,y) + u(x-h,y) + u\Big(x-\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) \\
&\qquad + u\Big(x+\frac{h}{2}, y+\frac{\sqrt{3}}{2}h\Big) + u(x+h,y) \\
&\qquad + u\Big(x+\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big) + u\Big(x-\frac{h}{2}, y-\frac{\sqrt{3}}{2}h\Big)\Bigg) \\
&= -\frac{1}{80640\sqrt{3}}\Big(189h^4u^{(0,6)}(x,y) + 945h^4u^{(2,4)}(x,y) + 315h^4u^{(4,2)}(x,y) \\
&\qquad + 231h^4u^{(6,0)}(x,y) + 7560h^2u^{(0,4)}(x,y) \\
&\qquad + 15120h^2u^{(2,2)}(x,y) + 7560h^2u^{(4,0)}(x,y) \\
&\qquad + 120960u^{(0,2)}(x,y) + 120960u^{(2,0)}(x,y)\Big) \\
&- \frac{c^2}{645120\sqrt{3}}\Big(h^2\big(7560h^2u^{(0,4)}(x,y) + 15120h^2u^{(2,2)}(x,y) \\
&\qquad + 7560h^2u^{(4,0)}(x,y) + 120960u^{(0,2)}(x,y) \\
&\qquad + 120960u^{(2,0)}(x,y)\big) + 967680u(x,y)\Big) \\
&- \frac{\alpha_2c^2}{483840\sqrt{3}}\Big(h^2\big(7560h^2u^{(0,4)}(x,y) + 15120h^2u^{(2,2)}(x,y) \\
&\qquad + 7560h^2u^{(4,0)}(x,y) + 120960u^{(0,2)}(x,y) \\
&\qquad + 120960u^{(2,0)}(x,y)\big) + 725760u(x,y)\Big) + \mathcal{O}(h^5). \qquad (4.7)
\end{aligned}
$$

Since

$$
-\frac{120960u^{(0,2)}(x,y) + 120960u^{(2,0)}(x,y)}{80640\sqrt{3}} - \frac{967680u(x,y)}{645120\sqrt{3}} = 0,
$$

Eq. (4.7) becomes

$$
\begin{aligned}
\tau(x,y) = -\frac{1}{80640\sqrt{3}}&\big(189h^4u^{(0,6)}(x,y) + 945h^4u^{(2,4)}(x,y) + 315h^4u^{(4,2)}(x,y) \\
&+ 231h^4u^{(6,0)}(x,y) + 7560h^2u^{(0,4)}(x,y) \\
&+ 15120h^2u^{(2,2)}(x,y) + 7560h^2u^{(4,0)}(x,y)\big)
\end{aligned}
$$

$$
\begin{aligned}
&- \frac{c^2h^2}{645120\sqrt{3}}\big(7560h^2u^{(0,4)}(x,y) + 15120h^2u^{(2,2)}(x,y) + 7560h^2u^{(4,0)}(x,y) \\
&\qquad + 120960u^{(0,2)}(x,y) + 120960u^{(2,0)}(x,y)\big) \\
&- \frac{\alpha_2c^2}{483840\sqrt{3}}\Big(h^2\big(7560h^2u^{(0,4)}(x,y) + 15120h^2u^{(2,2)}(x,y) \\
&\qquad + 7560h^2u^{(4,0)}(x,y) + 120960u^{(0,2)}(x,y) + 120960u^{(2,0)}(x,y)\big) \\
&\qquad + 725760u(x,y)\Big) + \mathcal{O}(h^5). \qquad (4.8)
\end{aligned}
$$

Substituting the exact solution

$$u(x,y) = \sin\big(c\cos(\theta)x + c\sin(\theta)y\big)$$

(note that more general solutions can be chosen) into (4.8) gives

$$
\begin{aligned}
\tau(x,y) = -\frac{1}{322560\sqrt{3}}c^2\Big(&35c^5h^5\cos\big(cx\cos(\theta) + cy\sin(\theta) + \theta\big) \\
&+ 21c^5h^5\cos\big(cx\cos(\theta) + cy\sin(\theta) + 3\theta\big) \\
&+ 7c^5h^5\cos\big(cx\cos(\theta) + cy\sin(\theta) + 5\theta\big) \\
&+ c^5h^5\cos\big(cx\cos(\theta) + cy\sin(\theta) + 7\theta\big) \\
&+ 35c^5h^5\cos\big(-cx\cos(\theta) - cy\sin(\theta) + \theta\big) \\
&+ 21c^5h^5\cos\big(-cx\cos(\theta) - cy\sin(\theta) + 3\theta\big) \\
&+ 7c^5h^5\cos\big(-cx\cos(\theta) - cy\sin(\theta) + 5\theta\big) \\
&+ c^5h^5\cos\big(-cx\cos(\theta) - cy\sin(\theta) + 7\theta\big) \\
&+ 5040\alpha_2c^4h^4\sin\big(cx\cos(\theta) + cy\sin(\theta)\big) \\
&+ 2940c^4h^4\sin\big(cx\cos(\theta) + cy\sin(\theta)\big) \\
&- 42c^4h^4\sin\big(cx\cos(\theta) + cy\sin(\theta) + 6\theta\big) \\
&+ 42c^4h^4\sin\big(-cx\cos(\theta) - cy\sin(\theta) + 6\theta\big) \\
&- 80640\alpha_2c^2h^2\sin\big(cx\cos(\theta) + cy\sin(\theta)\big) \\
&- 30240c^2h^2\sin\big(cx\cos(\theta) + cy\sin(\theta)\big) \\
&+ 483840\alpha_2\sin\big(cx\cos(\theta) + cy\sin(\theta)\big) + \mathcal{O}(h^5)\Big). \qquad (4.9)
\end{aligned}
$$

The last $6$ terms in (4.9) can be written in the form

$$\sin\big(cx\cos(\theta) + cy\sin(\theta)\big)\big(c^6h^4C_2 + c^4h^2C_1\big), \qquad (4.10)$$

where

$$
\begin{aligned}
C_1 &= \frac{1}{322560\sqrt{3}}\left(30240 - \frac{483840\alpha2}{c^2h^2}\right), \\
C_2 &= \frac{1}{322560\sqrt{3}}\left(-2940 + \frac{80640\alpha2}{c^2} + 84\cos(6\theta)\right).
\end{aligned}
$$

We have shown that the contribution of the RFB method to the stabilization of the Galerkin method is very poor, because the contribution from the bubble functions is small at least for $ch < 3$. The working principle of the RFB method depends on an automatic adaptation of the bubble functions. When the problematic parameter of the problem under consideration (the Peclet number for the convection-diffusion equation, the wavenumber for the Helmholtz problem) gets larger, the contribution from the bubble functions must be well adjusted. Here, we see that this adaptation is not well adjusted for the Helmholtz problem. In order to increase the contribution of the bubble functions (or to decrease the contribution if necessary) further without spoiling the automatic adaptation of the method, we modify the right-hand side of the bubble equations in (4.2) by multiplying with a constant, say $\mu$. Then, $\alpha_2$ becomes

$$\alpha_2 = \frac{2\mu c^2 h^2}{3(72 - c^2h^2)}.$$

After this modification, the bubble functions are no more residual-free. We call these modified functions as adapted-bubble functions. We call the piecewise-defined linear approximations to these adapted-bubble functions as pseudo-adapted bubble functions.

We give two examples here to validate the approach when $\theta = \pi/3$. Fig. 7 shows the graph of $C_1$ and $C_2$ when $\mu = 6.8$. It is clear that $C_1$ is decreased in magnitude substantially. It is almost zero when $ch$ is close to zero. There is not much change in $C_2$ in magnitude.

For the second example, consider $\alpha_2 = 0.0625c^2h^2$ which makes $C_1$ zero for all values of $ch$. In this case, the finite difference scheme in (4.6) is a fourth-order scheme

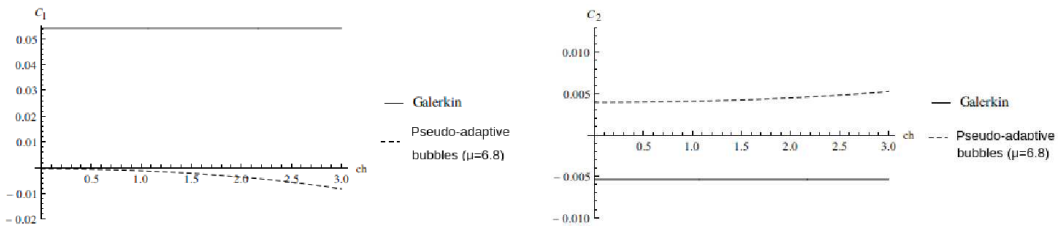


Figure 7: Comparison of the linear Galerkin method and the pseudo-adapted bubbles method ($\mu = 6.8$) for the coefficients $C_1$ (left) and $C_2$ (right) when $\theta = \pi/3$.

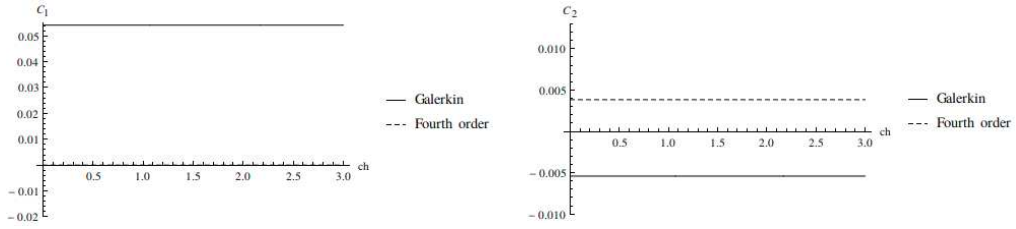


Figure 8: Comparison of the linear Galerkin method and the fourth-order accurate method for the coefficients $C_1$ (left) and $C_2$ (right) when $\theta = \pi/3$.

with seven points for plane waves. Fig. 8 shows the graph of $C_1$ and $C_2$ when $\alpha_2 = 0.0625c^2h^2$. While $C_1$ is zero for all values of $ch$, there is only a slight change in $C_2$ in magnitude. This fourth-order accurate finite difference scheme can be easily applied in triangular, trapezoidal and polygonal domains. Our main aim in this article is to propose adapted-bubbles approximated by standard Galerkin method. However, the above two methods will be used to do some comparison with the AB method.

## 5. Adapted-bubbles (AB) in 2D with triangular elements

We have shown using the pseudo-bubbles and the truncation error that the RFB method is not effective in 2D. However, it is possible to increase its accuracy with a simple modification, that is, multiplying the right-hand side of the bubble problems with a constant. We proposed two methods using this approach; a pseudo-adapted bubbles method and a fourth-order accurate finite difference scheme that uses seven points. However, our main aim is to obtain more accurate solutions by approximating the adapted-bubble functions with linear finite element method on a coarse mesh.

We use Algorithm 1 to find optimal values of $\mu$. We consider the problem in (4.1) for which $u = \sin(cy\sin(\theta) + cx\cos(\theta))$. Note that optimal values of $\mu$ changes with respect to $\theta$, and are same, for example, for $\theta = 0, \pi/3, 2\pi/3$ and for $\theta = \pi/6, \pi/2, 5\pi/6$. Because of this periodicity in $\mu$, we determine optimal values for $\theta = 0, \pi/6$ for which we expect maximum change in $\mu$. We choose a very coarse mesh by choosing $h = 1/5$ which corresponds to $N = 6$ nodes on each edges of the triangular domain. This gives $25$ equilateral triangular elements and a matrix of size $6 \times 6$. Each global mesh is discretized by choosing $N_s = 10$ nodes on each of the meshes (see Fig. 10). $N_s = 10$ amounts to $28 \times 28$ matrices on element level. We set the tolerance to $e_{tol} = 10^{-11}$ and maximum iteration to $i_{\max} = 200$.

We run the Algorithm 1 by choosing two initial guesses for $\mu$, say $\mu_1$ and $\mu_5$. We determine optimal values for $ch = 0.1, 0.3, 0.5, 0.5 + 0.1l$, $l = 1, \ldots, 45$ ($ch \in [0.1, 5]$), although it is possible to determine for larger values of $ch$. It is observed that, most of the time, the algorithm stopped because the iteration reached the $i_{\max}$. Although, for small values of $ch$ ($ch < 2$), the error decreased up to $10^{-8}$, for large values of $ch$ ($ch > 3$), the algorithm stopped when the error is around $10^{-4}$.

The optimal values of $\mu$ ($\mu_{vec}$) and the corresponding $ch$ values ($ch_{vec}$) are given in vector form in Appendix A. We use Matlab spline function to determine $\mu$ when $ch$ is between any of the two successive values in $ch_{vec}$, i.e., $\mu = spline(ch_{vec}, \mu_{vec}, ch)$. Fig. 9 shows cubic spline interpolation of $\mu$ for $\theta = 0, \pi/6$. It is clear that the difference is small for $ch < 1.5$. When $ch \approx 0.6$, the optimal value in one direction can be used in any other direction to get a good approximation. This $\theta$ independence is very important when an unstructured mesh is used and the solution is not a plane wave. Another observation is that the interpolation curves cut the $x$-axis which means that the optimal value of $\mu$ is zero at those points. In this case, the AB method is equivalent to the standard Galerkin method. For certain large values of $ch$, the standard Galerkin

**Algorithm 1:** Algorithm to find optimal values of $\mu$.

**Data:** $\theta$, $h$, $\mu_1$, $\mu_5$ ($\mu_5 > \mu_1$), $u$, $e_{tol}$, $i_{\max}$, $N_s$;
**Result:** $\mu$;
Initialization: $\mu_3 = (\mu_1 + \mu_5)/2$, $\mu_2 = (\mu_1 + \mu_3)/2$, $\mu_4 = (\mu_3 + \mu_5)/2$,
$e_i = \|u^h_{\mu_i} - u\|$, $i = 1, \ldots, 5$;
$itr = 1$;
**while** $\min(e_i) > e_{tol}$ **do**
  **if** $\min(e_i) = e_1$ **then**
    $h_\mu = (\mu_4 - \mu_1)/4$, $\mu_1 = \mu_1 - 2h_\mu$, $\mu_2 = \mu_1 + h_\mu$, $\mu_3 = \mu_1 + 2h_\mu$,
    $\mu_4 = \mu_1 + 3h_\mu$, $\mu_5 = \mu_1 + 4h_\mu$;
  **end**
  **if** $\min(e_i) = e_2$ **then**
    $h_\mu = (\mu_4 - \mu_1)/4$, $\mu_1 = \mu_1$, $\mu_2 = \mu_1 + h_\mu$, $\mu_3 = \mu_1 + 2h_\mu$,
    $\mu_4 = \mu_1 + 3h_\mu$, $\mu_5 = \mu_1 + 4h_\mu$;
  **end**
  **if** $\min(e_i) = e_3$ **then**
    $h_\mu = (\mu_4 - \mu_2)/4$, $\mu_3 = \mu_3$, $\mu_2 = \mu_3 - h_\mu$, $\mu_1 = \mu_2 - 5/2h_\mu$,
    $\mu_4 = \mu_3 + h_\mu$, $\mu_5 = \mu_3 + 2/3h_\mu$;
  **end**
  **if** $\min(e_i) = e_4$ **then**
    $h_\mu = (\mu_4 - \mu_1)/4$, $\mu_5 = \mu_5$, $\mu_4 = \mu_5 - h_\mu$, $\mu_3 = \mu_5 - 2h_\mu$,
    $\mu_2 = \mu_5 - 3h_\mu$, $\mu_1 = \mu_5 - 4h_\mu$;
  **end**
  **if** $\min(e_i) = e_5$ **then**
    $h_\mu = (\mu_4 - \mu_1)/4$, $\mu_5 = \mu_5 + 2h_\mu$, $\mu_4 = \mu_5 - h_\mu$, $\mu_3 = \mu_5 - 2h_\mu$,
    $\mu_2 = \mu_5 - 3h_\mu$, $\mu_1 = \mu_5 - 4h_\mu$;
  **end**
  $e_i = \|u^h_{\mu_i} - u\|$, $i = 1, \ldots, 5$;
  $\mu = \mu_3$;
  $itr = itr + 1$;
  **if** *itr* $= i_{\max}$ **then**
    break
  **end**
**end**

method gives accurate solutions. These values of $ch$ depend on $\theta$. For example, the standard Galerkin method gives a good approximation when $ch \approx 4.7165$ for $\theta = 0$.

To show effectiveness of the optimal values of $\mu$, we use the pseudo-adapted bubbles. For example, for $\mu = 5.4$ (when $ch \approx 0.6$), $\alpha_2 = 10.8c^2h^2/(3(72 - c^2h^2))$. Graphs of the coefficients $C_1$ and $C_2$ are provided in Fig. 11. It is obvious that both $C_1$ and $C_2$ are decreased in magnitude which is a verification that the AB method can mitigate the pollution effect substantially.

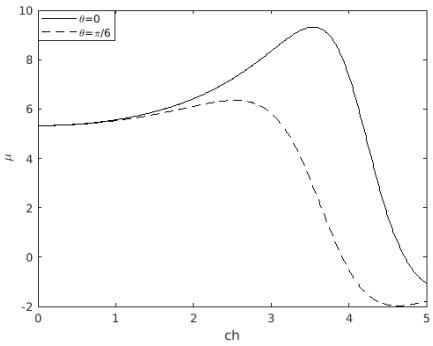


Figure 9: Cubic spline interpolation of $\mu$ when $\theta = 0, \pi/6$.

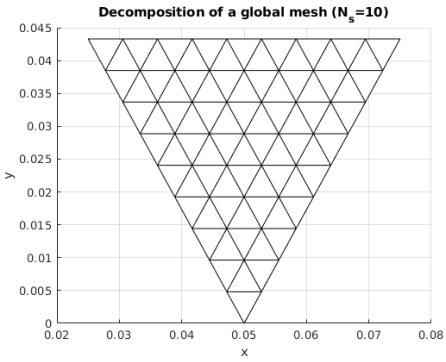


Figure 10: The decomposition of a global mesh with triangular elements when $N_s = 10$.

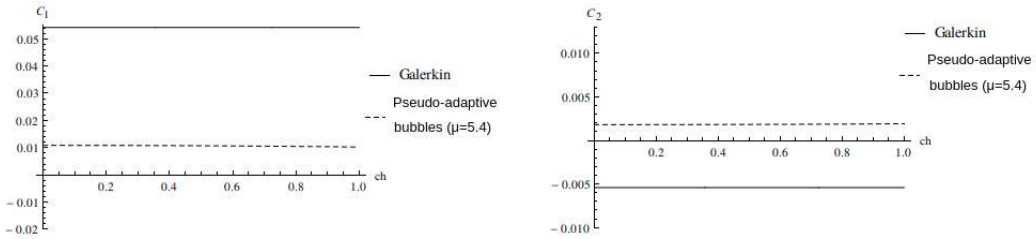


Figure 11: Comparison of the linear Galerkin method and the pseudo-adapted bubbles method ($\mu = 5.4$) for the coefficients $C_1$ (left) and $C_2$ (right) when $\theta = \pi/3$.

## 5.1. Numerical experiments

In this section, we provide numerical tests to asses the success of the AB method. We compare the AB method with the pseudo-adapted bubbles ($\mu = 6.8$) (PAB), RFB, fourth-order and standard linear finite element methods. We use standard linear finite element method to approximate the bubble functions for the AB and the RFB methods.

### 5.1.1. Numerical test 1

We consider the Helmholtz problem in (4.1). Equilateral triangular elements are used to decompose the domain. We consider cases $ch = 0.625, 1, 1.667$ when $\theta = 0$ for increasing wavenumber to compare the methods in mitigating the pollution effect. Fig. 12 shows the log-log plots of the error in infinity norm for $ch = 0.625$, $ch = 1$ and $ch = 1.667$. It is obvious that the AB method is better by far. The RFB method has very small contribution in stabilization of the standard Galerkin method. The PAB method outperforms the fourth-order scheme. Moreover, the pollution error for the pseudo-bubbles method and the fourth-order scheme is not negligible, particularly when $ch = 1, 1.667$. The AB method is pollution error free for $c > 40$. We observe pollution error for $c < 40$ but the error is small enough that the pollution error has almost no importance.

For larger values of $ch$ ($ch > 2$), only the AB method produces accurate results. That is why we consider only the AB method for $ch \geq 2$. Fig. 13 shows errors for the AB method when $ch = 2, 3, 4, 5$. Even in extreme cases ($ch > 3.5$), the AB method gives accurate results. To the best of author's knowledge, there exists no method in the literature that works in this regime in 2D.

We also test the AB method with values of $\mu$ obtained by $\mu = spline(ch_{vec}, \mu_{vec}, ch)$. To this end, we set the mesh size to $h = 1/40$ and report errors for the wavenumbers $c = 5 + 10l$, $l = 1, \ldots, 19$ for which $ch \in [0.375, 4.875]$. Fig. 14 shows the error for $\theta = 0, \pi/6$. The results show that the values obtained by the cubic spline interpolation gives accurate results.

### 5.1.2. Numerical test 2: Robin boundary condition and external source

We test the AB method when Robin boundary condition is imposed on a part of the boundary of the domain. We consider

$$\begin{cases} -\Delta u - c^2 u = \sin(x) & \text{in } \Omega, \\ u(x, y) = 0.1 & \text{on } \partial\Omega_D, \\ \dfrac{\partial u}{\partial \mathbf{n}} = iu & \text{on } \partial\Omega_R, \end{cases} \tag{5.1}$$

where $c = 20$, and $\Omega$, $\Omega_D$ and $\Omega_R$ are represented in Fig. 15 (left). As a reference solution, we get a solution using standard Galerkin method on a fine mesh where $40000$ uniform triangular elements are used for which $ch = 0.1$. Fig. 15 (right) shows the contour plot of the real part of the solution. We show contour plots of the real part of the solutions obtained by the AB method for $ch = 0.5, 1, 2$ in Fig. 16. We report the maximum and minimum values of the approximate solutions. Results show that the AB method shows the characteristics of the reference solution for all cases. This is important in application of the multigrid method as a solver.

### 5.1.3. Numerical test 3: L-shaped domain and a different triangulation

In this test problem, we change the domain and use a different triangulation. We use a L-shaped domain with the vertices $(-1, -1)$, $(-1, 1)$, $(1, 1)$, $(0, 1)$, $(0, 0)$ and $(-1, 0)$. To decompose the domain, the following Matlab code is used for which $ch \approx 0.625$.

$$\begin{cases} model = createpde(1); \\ geometryFromEdges(model, @lshapeg); \\ generateMesh(model,' GeometricOrder',' linear',' Hmax', \\ \qquad\qquad 0.625/c,' Hmin', 0.625/c). \end{cases} \tag{5.2}$$

The mesh for the case $c = 3.5\pi$ can be seen in Fig. 17. We consider the Dirichlet problem in (4.1) for $\theta = \pi/3$. Figs. 17 and 18 show the plots of the exact and approximate

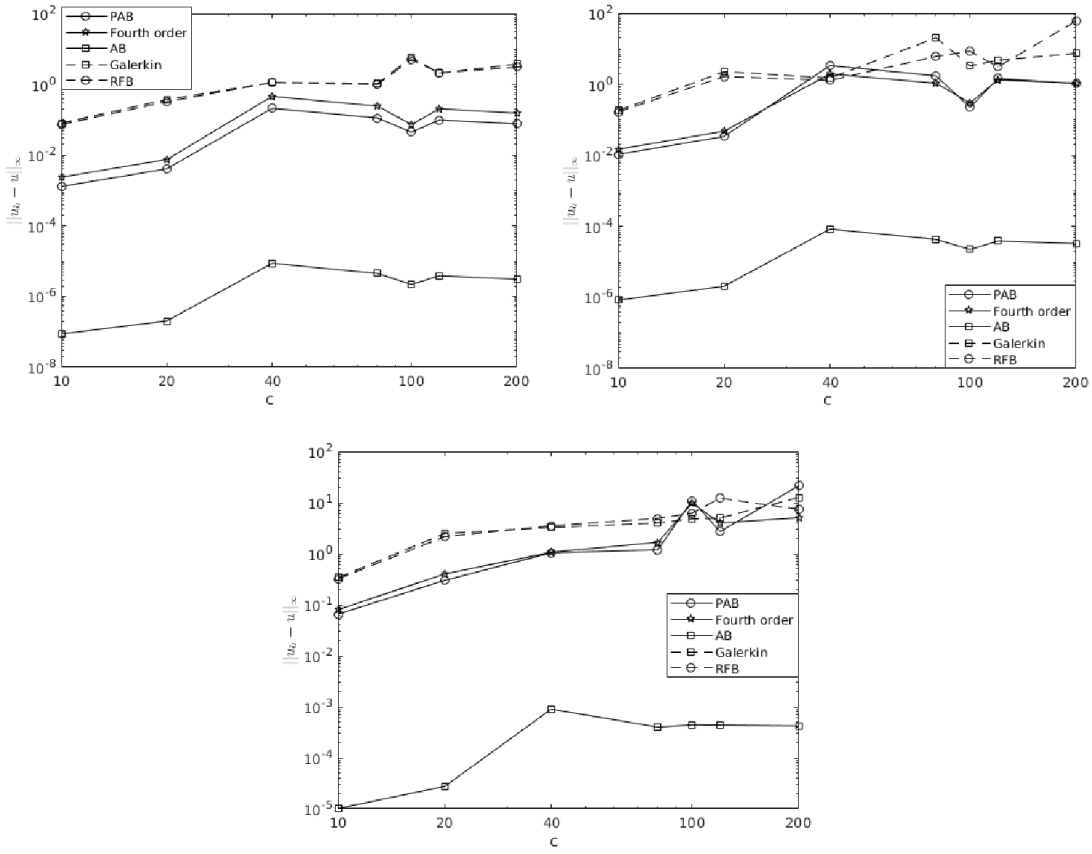


Figure 12: Comparison of the methods when $\theta = 0$ for $ch = 0.625$ (top left), $ch = 1$ (top right) and $ch = 1.667$ (bottom).

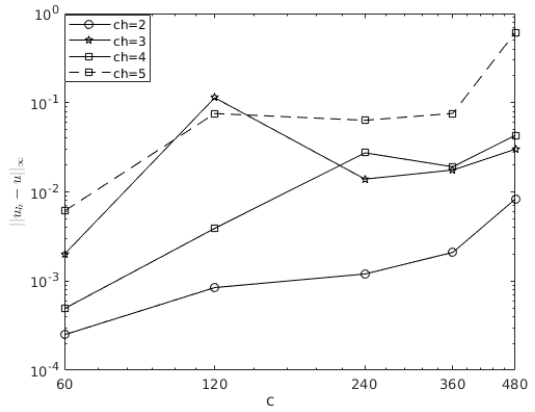


Figure 13: Error for the AB method for varying wavenumber when $ch = 2, 3, 4, 5$.

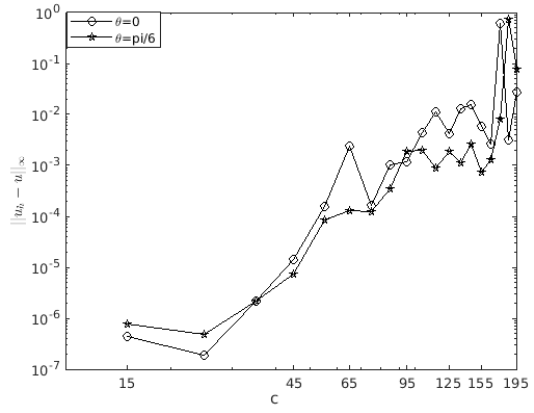


Figure 14: Error for the AB method for $\theta = 0, \pi/4$ for varying wavenumber and fixed $h = 1/40$ ($ch \in [0.375, 4.875]$).

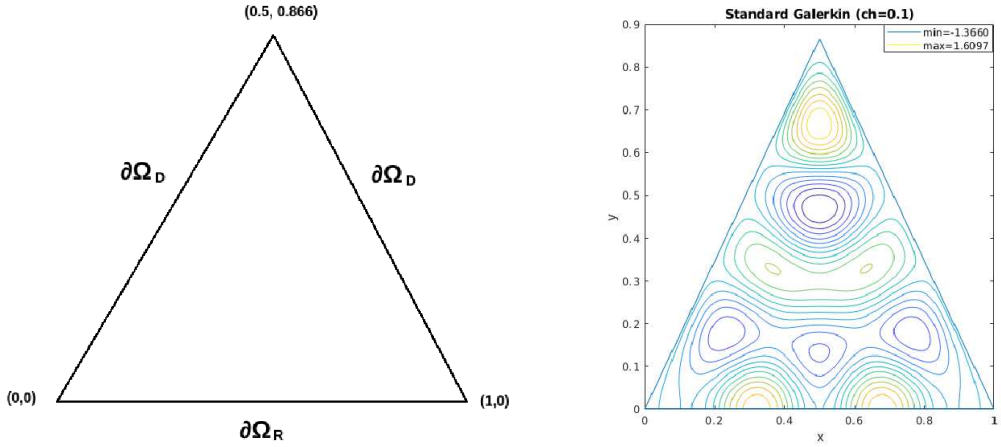


Figure 15: Problem configuration (left) and a reference solution (right) obtained with standard Galerkin method with 40000 uniform triangular elements for which $ch = 0.1$.

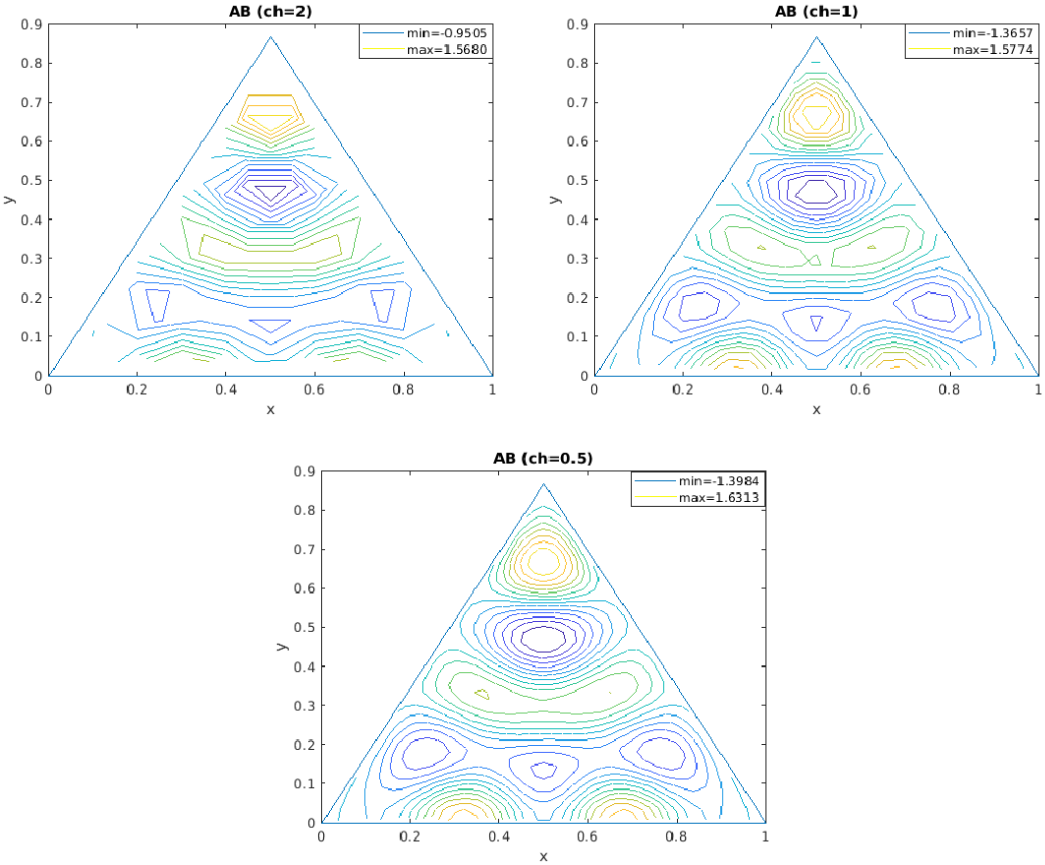


Figure 16: Contour plots of the approximate solutions obtained by the M-RFB method when $c = 20$ using different meshes ($ch = 0.5, 1, 2$).

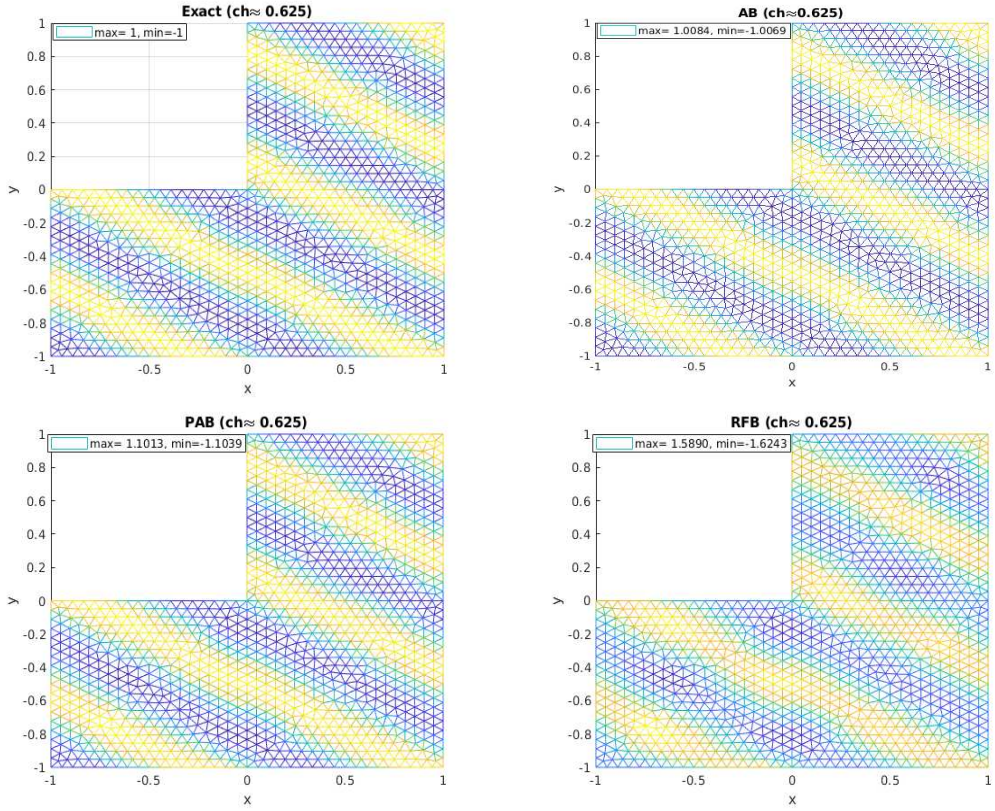


Figure 17: Plots of the exact and approximate solutions obtained by the AB, PAB and RFB methods when $c = 3.5\pi$.

solutions obtained by the AB, PAB and RFB methods for $c = 3.5\pi$ and $c = 16.5\pi$, respectively. We also report the maximum and minimum values of the approximate solutions on the graphs. Results show that the AB method is better by far especially for larger wavenumbers. Furthermore, we give the plots of the exact solution and approximate solution for AB given in Fig. 19 when $c = 33.5\pi$. We did not report solutions for the PAB and RFB methods as their results are no more related to the exact solution. The results show the effectiveness of the AB method when unstructured mesh is used.

### 5.1.4. Numerical test 4: A circular complex domain

In this test problem, we consider a complex domain which is obtained by the following Matlab code:

$$\begin{cases} model = createpde(1); \\ geometryFromEdges(model, @scatterg); \\ generateMesh(model,'GeometricOrder','linear','Hmax', \\ \qquad\qquad 0.625/c,'Hmin', 0.625/c). \end{cases} \tag{5.3}$$

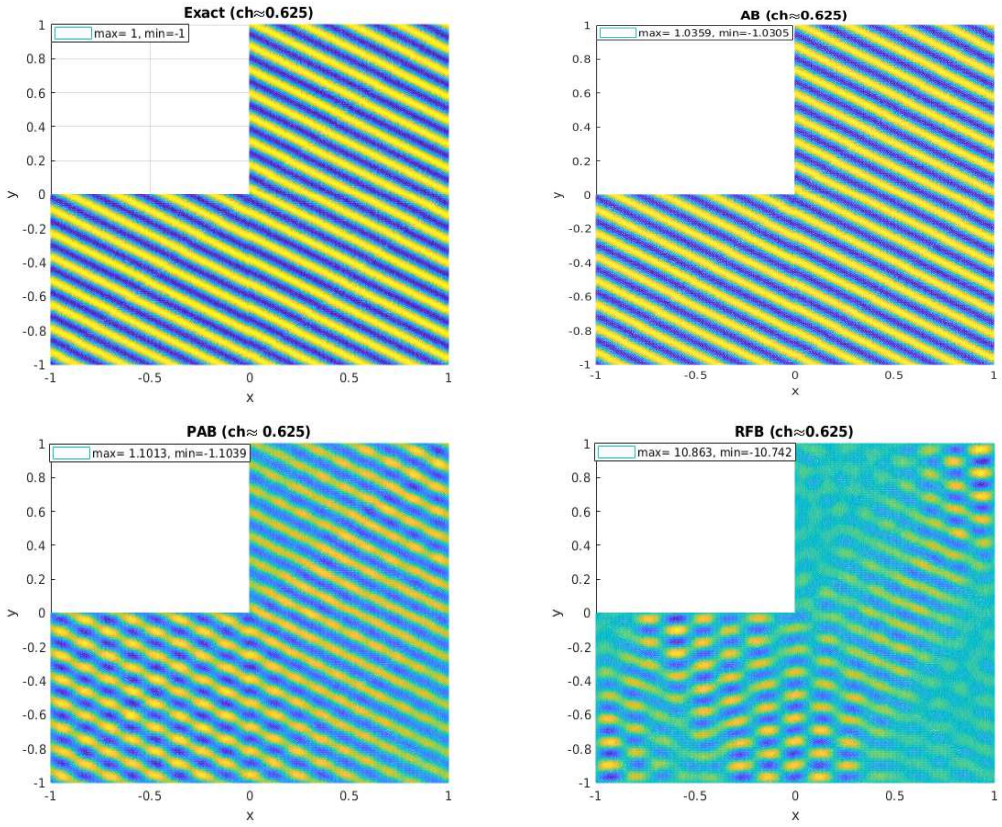


Figure 18: Plots of the exact and approximate solutions obtained by the AB, PAB and RFB methods when $c = 16.5\pi$.

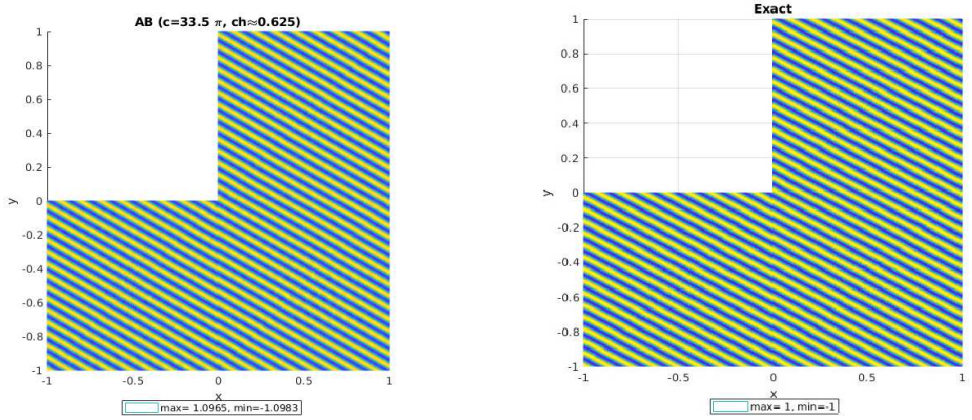


Figure 19: Contour plots of the approximate solutions obtained by the AB method and of the exact solutions on the same mesh when $c = 33.5\pi$.

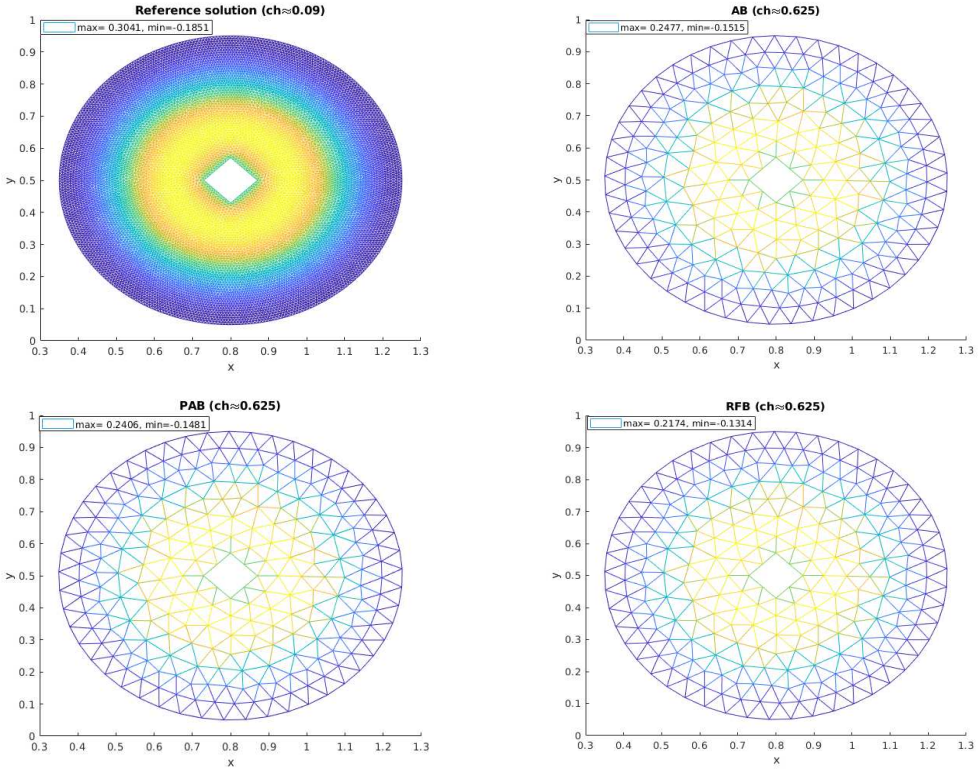


Figure 20: Plots of the reference solution and approximate solutions obtained by the AB, PAB and RFB methods when $c = 3.5\pi$.

While homogenous Neumann boundary condition is imposed on the outer boundary of the domain, i.e, $\partial u/\partial x = 0$, Dirichlet boundary condition is imposed on the inner boundary of the domain for which $u(x, y) = 0.1$. The right-hand side of the problem is set to zero. The reference solution is obtained by standard Galerkin method on a fine mesh for which $ch \approx 0.09$. While Fig. 20 shows the plots of the reference solution and approximate solutions of the AB, PAB and RFB method for $c = 3.5\pi$, Fig. 21 shows for $c = 16.5\pi$. We see that the RFB method is worst in any case. Although the AB and PAB give similar results for smaller wavenumbers, the AB method is better by far than the PAB for large wavenumbers. This and the previous tests show the success of the AB method on complex domains with unstructured meshes. We note that, the difference between the maximum of the reference solution and the maximum of the AB solution is not so small because a very fine mesh is used for the reference solution.

## 6. AB method with rectangular elements

Although the AB method is very effective with triangular elements, in some domains, rectangular elements may have some advantages such as in a rectangular re-

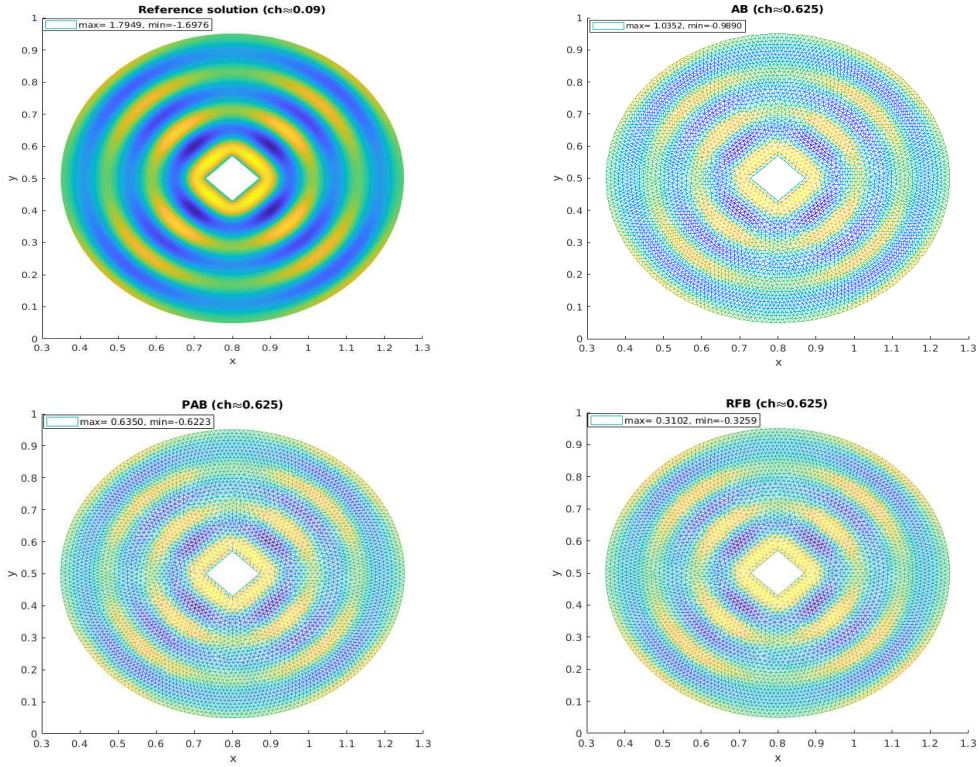


Figure 21: Plots of the reference solution and approximate solutions obtained by the AB, PAB and RFB methods when $c = 16.5\pi$.

gion. For a rectangular element, there are five bubble equations to be solved

$$\begin{cases} -\Delta\varphi_i - c^2\varphi_i = \mu c^2\psi_i & \text{in } K, \quad i = 1,\dots,4, \\ \varphi_i = 0 & \text{on } \partial K, \end{cases} \tag{6.1}$$

and

$$\begin{cases} -\Delta\varphi_f - c^2\varphi_f = f & \text{in } K, \\ \varphi_f = 0 & \text{on } \partial K, \end{cases} \tag{6.2}$$

where $\psi_i$, $i = 1,\dots,4$ are the bilinear basis functions of a rectangular element. We consider the Dirichlet problem in (4.1) on unit square. The optimal values of $\mu$ are same, for example, for $\theta = 0, \pi/2$, and for $\pi/4, 3\pi/4$. We expect to see the maximum difference in $\mu$ when $\theta = 0$ and $\theta = \pi/4$. That's why we report optimal values of $\mu$ for $\theta = 0, \pi/4$. To this end, we use the Algorithm 1. We set $h = 1/5$ which corresponds to $25$ square elements and a matrix of size $16 \times 16$. We set $N_s = 10$ which corresponds to matrices of size $64 \times 64$ on element level. We set the tolerance to $e_{tol} = 10^{-11}$ and the maximum iteration to $i_{\max} = 200$.

We run the Algorithm 1 by choosing two initial guesses for $\mu$, say $\mu_1$ and $\mu_5$. We determine optimal values for $ch = 0.1, 0.3, 0.5, 0.5 + 0.1l$, $l = 1,\dots,45$ ($ch \in [0.1, 5]$),

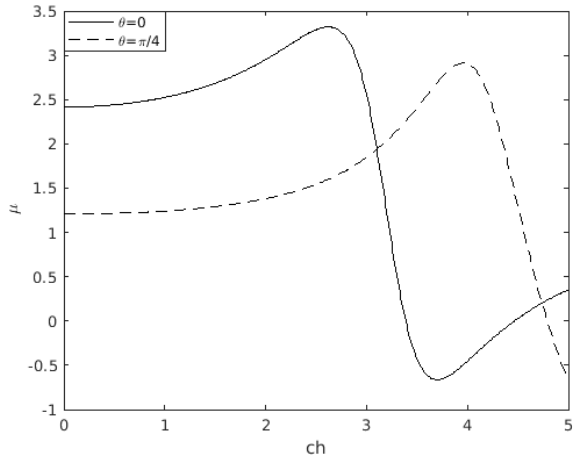


Figure 22: Cubic spline interpolation of $\mu$ when $\theta = 0, \pi/4$.

although it is possible to determine for larger values of $ch$. Convergence in $\mu$ is observed in Algorithm 1 for all cases.

The optimal values of $\mu$ ($\mu_{vec}$) and the corresponding $ch$ values ($ch_{vec}$) are given in vector form in Appendix A. We use Matlab spline function to determine $\mu$ when $ch$ is between any of the two successive values in $ch_{vec}$, i.e., $\mu = spline(ch_{vec}, \mu_{vec}, ch)$. Fig. 22 shows cubic spline interpolation of $\mu$ for $\theta = 0, \pi/4$. It is clear that the difference is not small even for small values of $ch$. This shows that rectangular elements are more sensitive in the change of $\theta$ and this makes the method less efficient when an unstructured mesh is used or the solution is not a plane wave. As observed for the triangular element, the interpolation curves cut the $x$-axis which means that the optimal value of $\mu$ is zero at those points. In this case, the AB method is equivalent to the standard Galerkin method. For example, for $ch = 4.790539642432885$, the standard Galerkin method produces error in infinity norm less than $10^{-8}$ when $\theta = \pi/4$ and $c < 1000$.

## 6.1. Numerical experiment

To see the performance of the AB method with rectangular elements, we consider the Helmholtz problem in (4.1) on unit square for which the exact solution is

$$u = \sin\big(cy\sin(\theta) + cx\cos(\theta)\big).$$

We compare the AB method with two higher order methods from the literature. We consider the sixth-order finite difference schemes $\mathrm{H}^{Wu}$ proposed in [28] and $\mathrm{H}^{asmpt}$ proposed in [5]. We note that $\mathrm{H}^{asmpt}$ is designed for small $h$ and the parameters of $\mathrm{H}^{Wu}$ are provided up to $ch = 2.5$. That is why we compare the three methods when $ch \leq 2.5$. Fig. 23 shows the error of the methods in infinity-norm for varying $c$ when

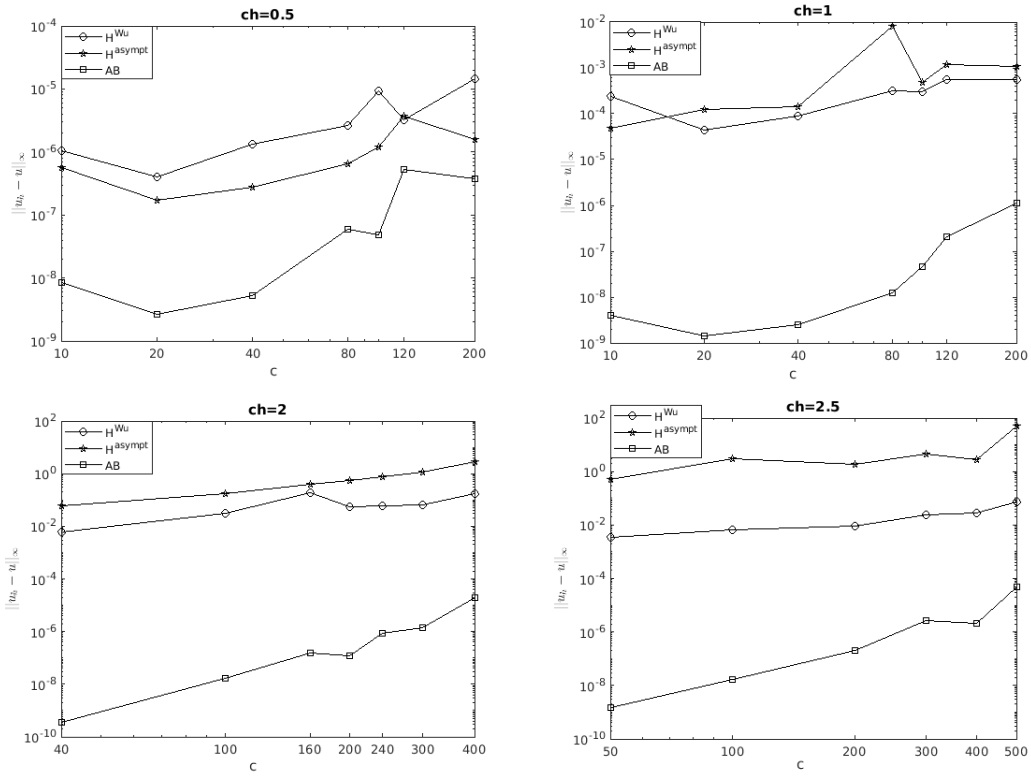


Figure 23: Comparison of the AB method with $\mathsf{H}^{Wu}$ and $\mathsf{H}^{asmpt}$ for different values of $ch$ when $\theta = \pi/4$.

$ch = 0.5, 1, 2, 2.5$ and $\theta = \pi/4$. It is obvious that the AB methods is better by far. The error for the AB method is so small that the pollution effect observed has no importance.

Furthermore we report error for the AB method for $ch = 3, 4, 5$ in Fig. 24 when $\theta = \pi/4$. Results show that, the AB method performs very well in this regime. Finally, we test the AB method with the values of $\mu$ obtained by $\mu = spline(ch_{vec}, \mu_{vec}, ch)$. To this end, we set the mesh size to $h = 1/40$ and report errors for the wavenumbers $c = 5 + 10l$, $l = 1, \ldots, 19$ for which $ch \in [0.375, 4.875]$. Fig. 25 shows the error for $\theta = 0, \pi/4$. The results show that the cubic spline interpolation gives accurate values of $\mu$. The error deteriorates as $ch$ increases which is related to the oscillatory behavior of $\mu$ for large $ch$. It is possible to decrease the error for large $ch$ by using more points for the interpolation, that is, by applying the Algorithm 1 for more points.

## 6.2. Comparison of triangular and rectangular elements

We considered triangular and rectangular elements for the AB method. Both of them has some advantages and disadvantages. We report our observations as follows:

- Rectangular elements use $9$ points per degrees of freedom but triangular elements use $7$ points per degrees of freedom.

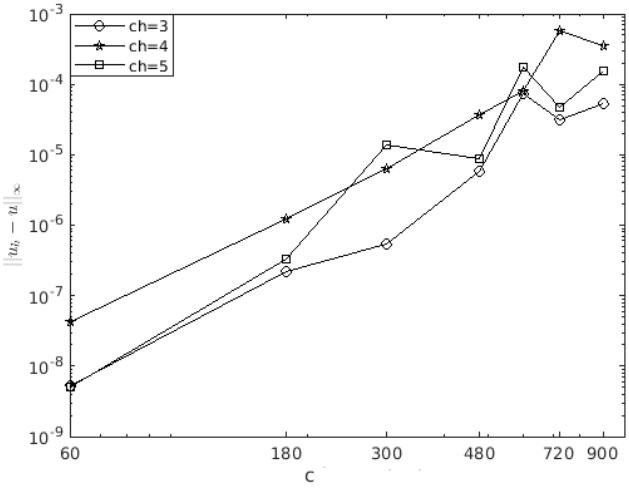


Figure 24: Error for the AB method for $ch = 3, 4, 5$ when $\theta = \pi/4$.

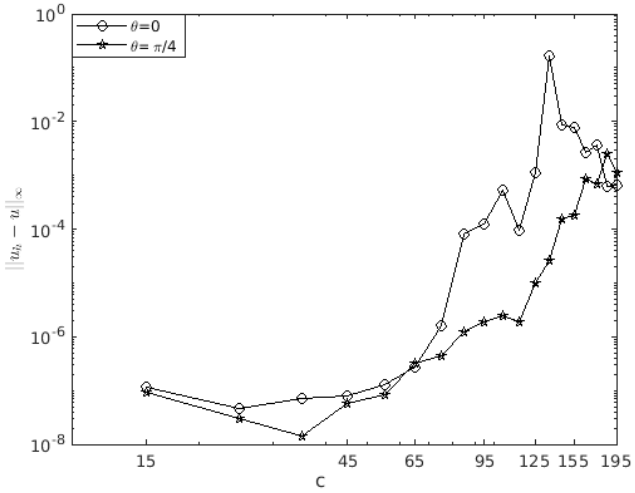


Figure 25: Error for the AB method for $\theta = 0, \pi/4$ for varying wavenumber and fixed $h = 1/40$ ($ch \in [0.375, 4.875]$).

- While a rectangular element requires solving $5$ different bubble equations, a triangular element requires $4$. This makes rectangular elements less efficient when nonuniform mesh is used.
- Square elements allows to get high accuracy but it is very sensitive to change in $\theta$.
- The optimal values of $\mu$ does not change much for $ch < 1.5$ when triangular element is used. That is why triangular element is more suitable for the unstructured mesh and when the solution is not a plane wave.

## 7. Conclusion

In this article, we proposed an adapted-bubble (AB) approach for the Helmholtz equation in 2D. The RFB method requires obtaining the bubble functions which is generally as difficult as solving the original problem. We showed that this is not the case for the Helmholtz problem. The standard Galerkin finite element method can be used to obtain approximations to the bubble functions. In other words, the bubbles method does not depend on another stabilized method when applied to the Helmholtz problem. We showed that the contribution of the RFB method in stabilization of the standard Galerkin method is not well adjusted. We modified the RFB method by multiplying the right-hand-side of the bubble problems with a constant. We reported the optimal values of this constant for equilateral triangular elements and square elements. Various numerical experiments proved the robustness of the AB method in terms of the parameters provided.

Our observations suggest that triangular element is more suitable for the Helmholtz problem when the AB method is used. It is possible to get high accuracy with the square element but it is very sensitive to change in the direction of the plane wave. We compared the AB method with the successful methods in the literature and observed that the AB method is better by far. We have provided numerical experiments up to $ch = 5$ but it is possible to obtain optimal values of $\mu$ for larger values of $ch$.

## Appendix A. Optimal values of $\mu$ and the corresponding $ch$ values in vector form

$ch$ values in vector form

$$\begin{aligned} ch_{vec} = [&0.1, 0.3, 0.5, 0.6, 0.7, 0.8, 0.9, 1, \\ &1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, \\ &2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3, \\ &3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4, \\ &4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5]. \end{aligned}$$

### A.1. Triangular element

Optimal values of $\mu$ for $\theta = 0$ in vector form

$$\begin{aligned} \mu_{vec} = [&5.327719956636429, 5.345857808366418, 5.382489757058465, \\ &5.407989413148243, 5.438460109353652, 5.474075833312540, \\ &5.515041258856693, 5.561594612479935, 5.614009699855814, \\ &5.672598376894667, 5.737713328671049, 5.809750686848182, \\ &5.889152676913680, 5.976410280722448, 6.072064377792510, \end{aligned}$$

$$6.176706837486098, 6.290978290501110, 6.415563473225142,$$
$$6.551180696637541, 6.698563047949805, 6.858428554840047,$$
$$7.031430430385674, 7.218076259176510, 7.418616213852046,$$
$$7.632857896103749, 7.859905000960552, 8.097757478101812,$$
$$8.342762891406892, 8.588806053623557, 8.826266692960303,$$
$$9.040626218167997, 9.210963539016667, 9.308522944373376,$$
$$9.296281709894537, 9.130763798207044, 8.767823524773119,$$
$$8.173785970628526, 7.340184319019317, 6.296695488970727,$$
$$5.112970113754272, 3.885332948333235, 2.711151351594336,$$
$$1.665912898816168, 0.791456837192823, 0.097873480070120,$$
$$-0.426786552882999, -0.806363243821480, -1.068881017195638].$$

Optimal values of $\mu$ for $\theta = \pi/6$ in vector form

$$\mu_{vec} = [5.327500566840172, 5.343139684177004, 5.374658154648253,$$
$$5.396433582127344, 5.422252605472676, 5.452159150820080,$$
$$5.486181575597186, 5.524375915527344, 5.566683244270854,$$
$$5.613116296438420, 5.663668867743165, 5.718083836516598,$$
$$5.776230881609081, 5.837764252662964, 5.902346980738457,$$
$$5.969015282499461, 6.036898490972817, 6.104559127009047,$$
$$6.170519679550665, 6.231534246387810, 6.284398992545903,$$
$$6.325117751575419, 6.346443474292755, 6.341353135394456,$$
$$6.299870457964019, 6.210797187095579, 6.058884601581376,$$
$$5.829295393041615, 5.506995035713146, 5.079595044652093,$$
$$4.540263347327709, 3.893709744679224, 3.157825169735588,$$
$$2.364593647238507, 1.556573121109977, 0.779709045891650,$$
$$0.074739679694176, -0.529429289686959, -1.018631046636217,$$
$$-1.391723047488625, -1.658417819015876, -1.833041429519653,$$
$$-1.932390402728197, -1.971378095936671, -1.965993465721103,$$
$$-1.927450075536035, -1.864745485488590, -1.786318673686765].$$

## A.2. Square element

Optimal values of $\mu$ for $\theta = 0$ in vector form

$$\mu_{vec} = [2.416319465637207, 2.424534025788307, 2.441243074994782,$$
$$2.452954335795948, 2.467026924636252, 2.483582415618747,$$
$$2.502765497954248, 2.524745410494384, 2.549717507155262,$$

$$2.577904362618347, 2.609556141820019, 2.644949184912659,$$
$$2.684381580352784, 2.728163421712815, 2.776597498147749,$$
$$2.829944356950000, 2.888359931381492, 2.951786996740848,$$
$$3.019767937660218, 3.091124134766869, 3.163411221162533,$$
$$3.232005997160449, 3.288617167039775, 3.318991720657795,$$
$$3.299806300662458, 3.195715750195086, 2.960220706649126,$$
$$2.548427447285503, 1.949552637320012, 1.226098413064610,$$
$$0.511312862685882, -0.061222607278032, -0.430717079229362,$$
$$-0.613803158735936, -0.662595820163956, -0.628986020180263,$$
$$-0.551063317032930, -0.452924949705283, -0.348523051725027,$$
$$-0.24540103113369, -0.147328697469110, -0.055946590397818,$$
$$0.028265541656562, 0.105435481182771, 0.175988597167188,$$
$$0.240477321280367, 0.299489638392403, 0.353602075764519].$$

Optimal values of $\mu$ for $\theta = \pi/4$ in vector form

$$\mu_{vec} = [1.207990837097168, 1.210443985462189, 1.21551946040581,$$
$$1.219085277147315, 1.223379621573460, 1.228445018359571,$$
$$1.234332747931739, 1.241103587969155, 1.248829170500540,$$
$$1.257593345179885, 1.267493975741763, 1.278644993610043,$$
$$1.291178923920961, 1.305249860696495, 1.321037051104940,$$
$$1.338749134878162, 1.358629235798435, 1.380960969906300,$$
$$1.406075562717160, 1.434360164776444, 1.466267490421888,$$
$$1.502326714573428, 1.543155395373466, 1.589471720099695,$$
$$1.642105573183308, 1.702005448502853, 1.770235638371444,$$
$$1.847953597258644, 1.936349648658530, 2.036518372245463,$$
$$2.149210682581971, 2.274386032942629, 2.410448411866114,$$
$$2.553027020898201, 2.693212894521038, 2.815423081095785,$$
$$2.895743076157467, 2.902747395841447, 2.803614147499320,$$
$$2.576555057903004, 2.224515015732089, 1.779434688475981,$$
$$1.291013929040052, 0.807195796274921, 0.360445432500303,$$
$$-0.035147079326975, -0.379057482114488, -0.677965467522639].$$

## References

[1] G. B. Alvarez, A. F. D. Loula, E. G. Dutra do Carmo, and F. A. Rochinha, *A discontinuous finite element formulation for Helmholtz equation*, Comput. Methods Appl. Mech.

Engrg. 195 (2006), 4018–4035.
[2] I. M. BABUŠKA AND S. A. SAUTER, *Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?*, SIAM J. Numer. Anal. 34 (1997), 2392–2423.
[3] F. BREZZI AND A. RUSSO, *Choosing bubbles for advection-diffusion problems*, Math. Models Methods Appl. Sci. 4 (1994), 571–587.
[4] P.-H. COCQUET AND M. J. GANDER, *How large a shift is needed in the shifted Helmholtz preconditioner for its effective inversion by multigrid?*, SIAM J. Sci. Comput. 39 (2017), A438–A478.
[5] P.-H. COCQUET, M. J. GANDER, AND X. XIANG, *Closed form dispersion corrections including a real shifted wavenumber for finite difference discretizations of 2D constant coefficient Helmholtz problems*, SIAM J. Sci. Comput. 43 (2021), A278–A308.
[6] S. CONGREVE, P. HOUSTON, AND I. PERUGIA, *Adaptive refinement for $hp$-version Trefftz discontinuous Galerkin methods for the homogeneous Helmholtz problem*, Adv. Comput. Math. 45 (2019), 361–393.
[7] H. C. ELMAN, O. G. ERNST, AND D. P. O'LEARY, *A multigrid method enhanced by Krylov subspace iteration for discrete Helmhotz equations*, SIAM J. Sci. Comput. 23 (2001), 1291–1315.
[8] Y. A. ERLANGGA, C. W. OOSTERLEE, AND C. VUIK, *A novel multigrid based preconditioner for heterogeneous Helmholtz problems*, SIAM J. Sci. Comput. 27 (2006), 1471–1492.
[9] O. G. ERNST AND M. J. GANDER, *Why it is difficult to solve Helmholtz problems with classical iterative methods*, in: *Numerical Analysis of Multiscale Problems*, Lecture Notes in Computational Science and Engineering, Springer, 83 (2012), 325–363.
[10] X. FENG, Z. LI, AND Z. QIAO, *High order compact finite difference schemes for the Helmholtz equation with discontinuous coefficients*, J. Comput. Math. 29 (2011), 324–340.
[11] X. FENG AND H. WU, *Discontinuous Galerkin methods for the Helmholtz equation with large wave number*, SIAM J. Numer. Anal. 47 (2009), 2872–2896.
[12] L. P. FRANCA, C. FARHAT, A. P. MACEDO, AND M. LESOINNE, *Residual-free bubbles for the Helmholtz equation*, Internat. J. Numer. Methods Engrg. 40 (1997), 4003–4009.
[13] L. P. FRANCA AND A. P. MACEDO, *A two-level finite element method and its application to the Helmholtz equation*, Internat. J. Numer. Methods Engrg. 43 (1998), 23–32.
[14] L. P. FRANCA, A. NESLITURK, AND M. STYNES, *On the stability of residual-free bubbles for convection-diffusion problems and their approximation by a two-level finite element method*, Comput. Methods Appl. Mech. Engrg. 166 (1998), 35–49.
[15] M. J. GANDER, I. G. GRAHAM, AND E. A. SPENCE, *Applying GMRES to the Helmholtz equation with shifted Laplacian preconditioning: What is the largest shift for which wavenumber-independent convergence is guaranteed?* Numer. Math. 131 (2015), 567–614.
[16] M. J. GANDER, F. MAGOULÈS, AND F. NATAF, *Optimized Schwarz methods without overlap for the Helmholtz equation*, SIAM J. Sci. Comput. 24 (2002), 38–60.
[17] M. J. GANDER AND H. ZHANG, *A class of iterative solvers for the Helmholtz equation: Factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized Schwarz methods*, SIAM Rev. 61 (2019), 3–76.
[18] F. IHLENBURG AND I. M. BABUŠKA, *Finite element solution of the Helmholtz equation with high wave number. I. The $h$-version of the FEM*, Comput. Math. Appl. 30 (1995), 9–37.
[19] S. KIRKUP, *The boundary element method in acoustics: A survey*, Appl. Sci. 9 (2019).
[20] A. I. NESLITURK, *Approximating the incompressible Navier-Stokes equations using a two-level finite element method*, Ph.D. Thesis, University of Colorado at Denver, ProQuest LLC, 1999.

[21] I. PERUGIA, P. PIETRA, AND A. RUSSO, *A plane wave virtual element method for the Helmholtz problem*, ESAIM Math. Model. Numer. Anal. 50 (2016), 783–808.
[22] A. SENDUR AND A. I. NESLITURK, *Applications of the pseudo residual-free bubbles to the stabilization of convection-diffusion-reaction problems*, Calcolo 49 (2012), 1–19.
[23] A. SENDUR, A. NESLITURK, AND A. KAYA, *Applications of the pseudo residual-free bubbles to the stabilization of the convection-diffusion-reaction problems in 2D*, Comput. Methods Appl. Mech. Engrg. 277 (2014), 154–179.
[24] I. SINGER AND E. TURKEL, *High-order finite difference methods for the Helmholtz equation*, Comput. Methods Appl. Mech. Engrg. 163 (1998), 343–358.
[25] C. C. STOLK, *A rapidly converging domain decomposition method for the Helmholtz equation*, J. Comput. Phys. 241 (2013), 240–252.
[26] C. C. STOLK, M. AHMED, AND S. K. BHOWMIK, *A multigrid method for the Helmholtz equation with optimized coarse grid corrections*, SIAM J. Sci. Comput. 36 (2014), A2819–A2841.
[27] T. STROUBOULIS, I. M. BABUŠKA, AND R. HIDAJAT, *The generalized finite element method for Helmholtz equation: theory, computation, and open problems*, Comput. Methods Appl. Mech. Engrg. 195 (2006), 4711–4731.
[28] T. WU AND R. XU, *An optimal compact sixth-order finite difference scheme for the Helmholtz equation*, Comput. Math. Appl. 75 (2018), 2520–2537.