Advanced Mathematical Models & Applications
Vol.7, No.3, 2022, pp.306-322

POLYNOMIAL APPROXIMATION OF AN INVERSE CAUCHY PROBLEM FOR HELMHOLTZ TYPE EQUATIONS*

Fatima Aboud[1†], Ibtihal Th. Jameel[1], Athraa F. Hasan[1], Baydaa Kh. Mostafa[1], Abdeljalil Nachaoui[2]

[1]College of Science,Diyala University, Diyala, Iraq
[2]Laboratoire de Mathematiques Jean Leray, Nantes Université, Nantes, France

Abstract. The objective of this paper is to solve numerically a Cauchy problem defined on a two-dimensional domain occupied by a material satisfying the Helmholtz type equations and verifying additional Cauchy-type boundary conditions on the accessible part of the boundary. A meshless numerical method using an approximation of the solution based on the polynomial expansion is applied. To confirm the efficiency of the proposed method, different examples were considered and the obtained linear system was solved using the well-known CG and CGLS algorithms.

Keywords: Helmholtz equation, inverse Cauchy Problem, meshless method, CG,CGLS.
AMS Subject Classification: 65N21, 41A10, 65F22, 65N35.
[†]**Corresponding author:** Fatima, Aboud, Diyala University, Diyala, Iraq, e-mail: *Fatima.Aboud@uodiyala.edu.iq*
Received: 1 July 2022; Revised: 9 August 2022; Accepted: 12 September 2022;
Published: 15 December 2022.

1 Introduction

This paper is concerned with the Cauchy problem for the Helmholtz equation. Helmholtz-like equations can arise naturally from physical applications related to wave propagation (Wang et al., 2020), vibrational phenomena, and heat transfer (Beskos, 1997), in the acoustic cavity problem Chen & Wong (1998), the radiation wave (Harari et al., 1998) and the heat conduction in fins (Kraus et al., 2001).

The Cauchy problem, (Regińska & Regiński, 2006; Ellabib & Nachaoui, 2008; Fu et al., 2009; Bergam et al., 2019; Chakib et al., 2019;Wang et al., 2020; Ellabib et al., 2021, 2022; Juraev & Gasimov, 2022), is one of the examples of inverse problems, (Huang & Chen, 2000; Boulkhemair et al., 2013; Lavrentiev, 2013; Isakov, 2017; Kozlov et al., 2018; Aboud et al., 2021; Nachaoui et al., 2021, 2022; Ouaissa et al., 2022). For this kind of Cauchy problems, some boundary conditions are given on a part of the boundary while no data is available on the rest of this boundary and the objective is to reconstruct the missing data from additional measurements on the accessible part of the boundary (Essaouini & Nachaoui., 2004; Mukanova B., 2013; Nachaoui & Salih, 2021; Reddy et al., 2021). It is well known that the Cauchy problem is ill-posed in the sense of Hadamard (Hadamard J., 1953; Lavrentiev, 2013; Dvalishvili et al., 2017). Therefore, an appropriate algorithm, which allows to circumvent this ill-posedness phenomenon, is necessary in order to solve in a stable way this kind of inverse problems.

*__How to cite (APA)__: Aboud, F., Jameel, I.Th., Hasan, A.F., Mostafa, B.Kh. & Nachaoui A. (2022). Polynomial approximation of an inverse Cauchy problem for Helmholtz type equations. *Advanced Mathematical Models & Applications*, *7*(3), 306-322.

In addition to the fact that the Cauchy problem is ill-posed, another difficulty specific to the Helmholtz equation is added. Indeed several works raised the difficulty of the numerical approximation of this equation when the wave number is large (Ihlenburg & Babuska, 1995, 1997).

In the last two decades, several methods have been proposed for solving the Cauchy problem for the Helmholtz equation (Yarmukhamedov, 2003; Regińska & Regiński, 2006; Fu et al., 2009; Qian and al., 2010; Berntsson et al., 2014, 2017; Huang et al., 2017; Qian & Feng, 2017; Yang et al., 2019; Wang et al., 2020, 2021).

Some of these methods propose algorithms that overcome this difficulty related to the magnitude of the wave number (Berntsson et al., 2014, 2017; Qian & Feng, 2017). These iterative methods depend on certain heuristic parameters whose choice ensuring convergence is not automatic. Also, it is not clear how large the wave number can be taken without the convergence being impaired. Recently, (Berdawood et al., 2021, 2022), propose new efficient alternating algorithms based on idea initially proposed in Jourhmane & Nachaoui (1999); Nachaoui, Aboud & Nachaoui (2021) to solve the Cauchy problem for the Poisson equation. They prove the convergence of the proposed procedures, for all values of wave number in the case of the Helmholtz equation and they show that their method can accelerate convergence in the case of the modified Helmholtz equation (Berdawood et al., 2020).

The main advantage of this approach is that there are no heuristic parameters and all the parameters used are completely expressed according to the specified data. Moreover, they prove that for any value of wavenumber one can specify an interval of relaxation parameter in which convergence is ensured and a sub-interval where convergence is very fast. The limits of these intervals are calculated according to the data. Unfortunately, in some cases the convergence acceleration interval is very small which makes it difficult to choose the relaxation parameter (Berdawood et al., 2022).

The main goal of this paper is to investigate a method depending on polynomial expansion to approximate the solution of the Cauchy problem for Helmholtz-type equation in a bounded domain. This method was proposed in Rasheed et al. (2021) to solve an inverse Cauchy problem for Poisson equation. It is a direct method which avoids the problem often encountered in iterative methods, namely the slowness of the algorithm. The major difficulty of this method is the transfer of the ill-posed character of the Cauchy problem to the matrix of the obtained linear system. This results in a very large number of conditions which quickly deteriorates the efficiency of the solver. We show that this inconvenience can be avoided by a preconditioning which is not very expensive.

In the following, we recall the inverse Cauchy problems for the Helmholtz equation.

2 Inverse Cauchy problems for the Helmholtz equation

We consider the inverse Cauchy problem for the Helmholtz equation defined by:

$$\begin{aligned} \Delta T + k^2 T &= F(x,y), \quad (x,y) \in \Omega && (1)\\ T(\rho,\theta) &= \tilde{T}(\theta), \quad 0 \le \theta \le \beta\pi && (2)\\ \partial_n T(\rho,\theta) \equiv \Phi(\rho,\theta) &= \widetilde{\Phi}(\theta), \quad 0 \le \theta \le \beta\pi, && (3)\end{aligned}$$

where $F(x,y), \tilde{T}(\theta)$ and $\widetilde{\Phi}(\theta)$ are sufficiently regular given functions, the boundary Γ of the domain $\Omega \subset \mathbb{R}^2$ is such that $\Gamma = \Gamma_1 \cup \Gamma_2$ with

$$\Gamma_1 = \{(r,\ \theta)\ :\ r\ =\ \rho(\theta)\ ,\ 0\ \le\ \theta\ <\beta\pi\} \tag{4}$$

and

$$\Gamma_2 = \{(r,\theta) : r = \rho(\theta), \beta\pi \le \theta < 2\pi,\ 0 < \beta < 2\} \tag{5}$$

Note that the normal derivative of $T(x, y)$ can be expressed in the following form (see (Liu & Kuo, 2018; Rasheed et al., 2021)),

$$\partial_n T = \eta(\theta)\left[cos(\theta) - \frac{\rho'}{\rho^2}sin(\theta)\right]\partial_x u + \eta(\theta)\left[sin(\theta) - \frac{\rho'}{\rho^2}cos(\theta)\right]\partial_y u \tag{6}$$

where $\eta(\theta)$ is given by

$$\eta(\theta) = \frac{\rho(\theta)}{\sqrt{\rho^2(\theta) + [\rho'(\theta)]^2}}. \tag{7}$$

3 Approximation of solution by a polynomial expansion

The idea is to approximate the solution T of (1)-(3) as a polynomial in the form:

$$T(x, y) = \sum_{i=1}^{m}\sum_{j=1}^{i} c_{ij}x^{i-j}y^{j-1} \tag{8}$$

where, $c_{\mathrm{ij}}, 1 \leq i \leq m$ and $1 \leq j \leq i$ are unknown coefficients to be determined.

Replacing in (2)-(3) the solution T by its approximation given in (8) and using the expression (6) of the normal derivative we obtain a system of equations whose evaluation on n_{1a} points of Γ_1 gives rise to $2 * n_{1a}$ equations. These equations combined with those obtained from the evaluation of equation (1), replacing T by its approximation (8), at n_{1b} points of Ω gives rise to a linear system of the form:

$$Ac = b \tag{9}$$

where c is a vector to be determined of length $n_2 = m(m+1)/2$ in which the coefficients c_{ij} have been reordered. The known data vector b has length $n_1 = 2 * n_{1a} + n_{1b}$ and A is a rectangular matrix of the system of order $n_1 \times n_2$.

Thus, solving the inverse Cauchy problem (1)-(3) is reduced to solving the system (9). It should be noted that in order to uniquely determine the solution of the system of linear algebraic equation (9), the number n_1 of collocation points and the number n_2 must satisfy the inequality $n_1 \geq n_2$.

4 Linear system and numerical method

4.1 Tikhonov's regularization

The matrix equation resulting from the polynomial approximation (9) is often highly ill-conditioned and the data for the Cauchy problem is usually not exact, it contains uncertainties. We therefore have to be particularly be careful when solving them. Most standard numerical methods struggle to achieve good precision in solving the system of linear algebraic equations (9) due to the large value of the condition number of the matrix A which increases considerably with the number of collocation points. The technique of regularization is a numerical method of treatment of the ill-posed discrete problem seeking to overcome the conditioning by replacing the matrix of the system of linear algebraic equations by a matrix having a better conditioning and whose solution of the associated system is close of the desired solution. This technique generally allows a better improvement of the numerical precision of the solution of the original problem but requires a good choice of regularization parameter for optimal performance.

We consider in our study only the regularization method of Tikhonov (the reader may consult Hansen (1998) for other regularization procedures).

Formally, the Tikhonov regularized solution of problem (9) is obtained as the solution of the regularized system

$$\left[A^T A + \alpha I\right] C_\alpha = A^T b \tag{10}$$

Note that if $\alpha = 0$ this system is equivalent to solving the following normal equation:

$$Dc = b_1, \tag{11}$$

where $b_1 = A^T b$ and $D = A^T A$. D is a symmetric positive definite matrix, thus the conjugate gradient (CG) method can be used to solve this last linear system. But in some cases, as we will see in the numerical results, this least squares solution will be completely dominated by the contributions of data errors and rounding errors. The addition of a regularization makes it possible to damp these contributions. Unfortunately even the regularization is not enough to circumvent the effect of the bad conditioning of these matrices, a preconditioning proves to be necessary.

4.2 Preconditioning

A **preconditioner** is a matrix with an appropriate advantage which can be used to obtain a new system having the same solution as the original one but which is better conditioned, i.e. whose number of conditions of the matrix resulting is smaller. We use the following kind of preconditioning: right, left and two-sided (as defined in Rasheed et al. (2021)).

The left and right preconditioning matrices P and Q are diagonal defined by:
$P = diag(p_1, p_2 \cdots p_n)$ and $Q = diag(q_1, q_2 \cdots q_n)$ where p_k and q_k are given by:

$$p_k = \gamma \left(\frac{\sum_{i=1}^{n} A_{i1}^2}{\sum_{i=1}^{n} A_{ik}^2} \right)^{\frac{1}{2}}, \qquad q_k = \delta \left(\frac{\sum_{i=1}^{n} A_{1i}^2}{\sum_{i=1}^{n} A_{ki}^2} \right)^{\frac{1}{2}} \tag{12}$$

for $k = 1, \cdots, n$. The parameters γ and δ are amplifying factors introduced to more reduce the condition number. If $\gamma = \delta = 1$, then the norm of each row and column of the obtained matrices QA and AP are the same respectively.

In the following we present the numerical results obtained from solving the linear system (9) by the CG and CGLS methods. In the CGLS, the matrix $D = A^T A$ is never calculated because this leads to unnecessary inaccuracies.

We show in our numerical experience that the CGLS is a good choice of solving the normal equations (10). It reduces the number of iterations and improves accuracy compared to CG.

5 Numerical results and discussion

In this section, we discuss the numerical results obtained using the polynomial approximation described in section (3) to solve the Cauchy problem for the Helmholtz equation in a two-dimensional bounded domain. The discrete system is solved using the Tikhonov regularization technique in conjunction with the CG and CGLS methods. The goal is to show that, in the case of the Cauchy problem governed by the Helmholtz equation, the procedure proposed here works well without restriction of the value of the wave number k. Contrary to some previous works, we will show, by a numerical analysis, that this method is stable, when the given data is noisy. In order to show the efficiency of the proposed procedure compared to other existing schemes concerned with the same problem, we treat examples of Berdawood et al. (2021, 2022).

5.1 Polynomial Examples

5.1.1 Examples with $u_{ex} = x^2 - y^2$

In this first example, we consider the domain bounded by $\rho(\theta) = 1$. The parameter β defining Γ_1 and Γ_1, see (4)-(5) is taken to be $\beta = 0.5$ The boundary conditions in (2)-(3) are computed from the exact solution $u_{\text{ex}} = x^2 - y^2$. The number of boundary collocation points used for discretizing the boundary is taken to be $\mathrm{n}_{1a} = 11$, and for the number of internal collocation points is $n_{1b} = 88$.

We are interested in the case where β**=0.5** which simulates the case where Cauchy data is available on a small part of the boundary. We examine the behavior of the method according to the wavenumber k.

We start by examining the impact of the tolerance variation in the stopping criterion of CG and CGLS on the quality of the solution.

First case: $Tol = 10^{-10}$.
The results for this case are given in Tables 1 - 4

Table 1: Accuracy and convergence for $k = \sqrt{15}$ with $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984480146	3	0.984480146
3	6	3.82617181E-13	6	3.53431456E-13
4	13	2.71076095E-12	13	1.57815691E-11
5	24	1.25102778E-11	23	1.57455066E-10
6	51	2.66719616E-09	50	4.58402391E-10
7	130	5.73363232E-09	104	3.26221713E-08
8	287	2.06856292E-08	238	1.48616206E-08
9	628	5.85053605E-03	468	5.85057346E-03
10	943	6.65922811E-03	546	7.04835449E-03

We observe from Tables 1 that the method gives rise to an accurate solution for $m \geq 3$. The best result (best precision with a very few number of iteration) is obtained for $m = 3$. This is normal, since we approximate a polynomial solution of degree 2 by a polynomial whose degree is $m - 1$. The best polynomial approximating a polynomial of degree q is a polynomial of the same degree. This allows us to conclude that if we take a relatively large m, the method ensures obtaining a very good approximation.

Table 2: Accuracy and convergence for $\sqrt{25.5}$ with $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984472074	3	0.984472074
3	6	3.38826241E-13	6	1.11068467E-12
4	13	1.21853710E-11	12	4.10071874E-11
5	24	1.00757311E-10	23	2.95448913E-10
6	50	9.43724405E-11	47	1.92213813E-09
7	115	1.29715217E-08	103	1.78820800E-09
8	291	1.78730096E-04	225	1.78770070E-04
9	601	1.95566917E-03	421	1.95562646E-03
10	789	3.73860718E-03	546	3.73881411E-03

The same observations drawn from Table 1 are valid for Table 2. This supports the conclusions made from the results of the first table.

As in Table 1 and Table 2, the same conclusions can be drown from Table 3. We also observe that the results are a little better than for $k = \sqrt{15}$ and $k = \sqrt{25.5}$.

The remark made from the results of Table 2, namely that the precision improves when k becomes larger and larger, is confirmed in the results of Table 4 . Unlike the iterative KMF method (see Berntsson et al. (2014); Berdawood et al. (2021, 2022)) we observe from the results

Table 3: Accuracy and convergence for $k = \sqrt{52}$ with $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984469932	3	0.984469932
3	6	2.71649722E-13	6	6.62061823E-13
4	12	3.00218846E-11	12	3.39699330E-11
5	22	3.17149310E-11	23	1.03545308E-10
6	50	4.13983795E-11	48	1.23093291E-11
7	118	2.70939616E-09	98	4.40119636E-09
8	231	1.75990931E-05	174	1.76158598E-05
9	475	3.53811957E-04	351	3.53726268E-04
10	624	6.21309895E-04	445	6.21277858E-04

presented in these last two tables that the method proposed here works well even when the wavenumber is large. We even observe a marked improvement in accuracy for these cases.

Table 4: Accuracy and convergence for $k = \sqrt{100}$ with $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984469527	3	0.984469527
3	6	4.09601773E-14	6	8.59588601E-13
4	12	4.69185779E-12	12	5.07247832E-11
5	22	3.73747172E-11	22	2.44003592E-10
6	50	7.54919212E-11	48	1.28764770E-12
7	94	8.78736779E-06	85	8.78878242E-06
8	182	7.04948035E-06	161	7.05266643E-06
9	418	3.65511550E-04	311	3.65687386E-04
10	423	1.25867992E-03	284	1.10680010E-03

From these first four results, we can conclude that, when the data comes from a polynomial function, the solution method based on the polynomial expansion produces a very good approximation even for large wavenumber. If the degree q of the polynomial forming the data is known, it suffices to take $m = q + 1$ to obtain a very accurate solution and in a small number of iterations in the solver of the linear system. We also conclude, from the results of Table 1 - 4 that $CGLS$ method is more accurate than CG method. It is also much faster when m becomes large, for example for the case $k = \sqrt{100}$ the iteration number is practically divided by 2 when m is large.

Second case : $Tol = 10^{-15}$

In Tables 5 - 8, we take the same data with $Tol = 10^{-15}$ for the stopping criteria.

For the tables 5 - 8, the same observations drawn from the first case are also valid for this case. This supports the conclusions made from the results of the first case. We observe also that for a tolerance Tol $= 10^{-15}$, we obtain a relative error of $\sim e^{-16}$, i.e. the calculated solution is an excellent approximation of the exact solution, when this solution is a polynomial. So, to summarize, we note that the method $CGLS$ is more precise than the CG and that even with equal precision $CGLS$ is faster.

5.1.2 Examples with $u_{ex} = 6x^2y^2 - x^4 - y^4$

In this second example, we study an example with a higher degree.

We consider the domain bounded by $\rho(\theta) = 0.5$. The parameter β defining Γ_1 and Γ_1, see (4)-(5) is taken to be $\beta = 0.5$ The boundary conditions in (2)-(3) are computed from the exact solution $u_{\text{ex}} = 6x^2y^2 - x^4 - y^4$.

Table 5: Accuracy and convergence for $k = \sqrt{15}$ with $\beta = 0.5$ and $Tol = 10^{-15}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984480146	3	0.984480146
3	7	9.12755322E-16	7	3.77980138E-16
4	14	3.78952645E-15	14	7.21601819E-16
5	29	1.16322812E-13	28	6.62657019E-15
6	68	6.04707192E-13	62	1.86258559E-14
7	158	2.33542657E-11	132	1.64513888E-12
8	349	4.70037373E-11	268	2.52040120E-13
9	2428	1.13099561E-08	1016	6.65894651E-10
10	14044	1.75054466E-05	3410	1.31807753E-05

Table 6: Accuracy and convergence for $k = \sqrt{25.5}$ with $\beta = 0.5$ and $Tol = 10^{-15}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984472074	4	0.984472074
3	7	1.44265899E-15	7	5.09868145E-16
4	14	5.58504934E-15	14	6.90729256E-16
5	29	7.88167679E-14	28	4.95019944E-15
6	62	2.66770643E-13	54	1.39751412E-14
7	155	1.06145736E-11	134	3.70890335E-14
8	427	5.05514191E-10	316	2.85704721E-14
9	2844	9.95416042E-08	1021	4.81938368E-10
10	15191	1.20139914E-05	3508	1.86095359E-06

Table 7: Accuracy and convergence for $k = \sqrt{52}$ with $\beta = 0.5$ and $Tol = 10^{-15}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984469932	3	0.984469932
3	7	9.59512661E-16	7	3.28408400E-16
4	14	3.00169388E-15	14	5.38313958E-16
5	29	6.01403759E-14	28	5.01986436E-15
6	58	1.20228077E-13	52	3.50558064E-15
7	149	8.80141791E-12	128	5.06843127E-13
8	473	8.19982392E-10	312	3.84308809E-12
9	3281	1.02000381E-06	1227	6.84986304E-10
10	5742	2.33372676E-05	1864	2.42305269E-05

In addition to the study made on the first example, we also study the impact of the number of collocation points.

First case: $n_{1a} = 11$**,** $n_{1b} = 88$ **and** $Tol = 10^{-10}$
In this case, the number of boundary collocation points used for discretizing the boundary is taken to be $n_{1a} = 11$, and for the number of internal collocation points is $n_{1b} = 88$. For both algorithms CG and CGLS, we take $Tol = 10^{-10}$ in the stopping criterion.

The results are given in Tables 9 - 12.

We note from Tables 9 - 12 that the accuracy is always good but deteriorates for small m. This is explained by the fact that, since the degree is high, a higher m is needed to have a better approximation. As the previos example, the best precision with a very few number of iteration is obtained for $m = 5$, since we approximate a polynomial solution of degree 4 by a polynomial

Table 8: Accuracy and convergence for $k = \sqrt{100}$ with $\beta = 0.5$ and $Tol = 10^{-15}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.984469527	4	0.984469527
3	7	2.83299759E-16	7	4.67295879E-16
4	14	9.90489913E-16	14	3.41366753E-16
5	29	3.07342662E-14	28	2.38961397E-15
6	58	3.03239428E-13	52	7.45233258E-15
7	156	1.79402653E-11	132	2.16095139E-13
8	514	4.07245841E-10	327	1.74280196E-12
9	3067	1.12306370E-07	1158	9.59299334E-10
10	7707	1.85625549E-04	2184	1.84165181E-04

Table 9: Accuracy and convergence for $k = \sqrt{15}$, with $\rho(\theta) = 0.5$, $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	0.998664828	3	0.998664828
3	7	1.00541144	7	1.00541144
4	14	1.377380975	14	1.377380975
5	32	1.90050451E-11	32	3.18677342E-12
6	69	6.10737313E-10	61	1.64776106E-10
7	152	4.20977703E-08	142	5.18793622E-11
8	466	1.46758312E-07	346	2.71289192E-08
9	1687	8.28351669E-04	942	8.24584976E-04
10	3707	8.45484978E-02	1906	8.46659492E-02

Table 10: Accuracy and convergence for $k = \sqrt{25.5}$, with $\rho(\theta) = 0.5$, $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.98618266E-01	3	9.98618266E-01
3	7	1.000024587	7	1.000024587
4	15	1.113095769	14	1.113095769
5	33	2.10793672E-10	32	3.68721217E-11
6	74	3.20771192E-10	65	9.52506593E-09
7	159	3.57762639E-09	145	2.97209365E-10
8	462	3.85260228E-07	393	2.38206922E-08
9	1747	1.57054331E-04	1032	1.52807113E-04
10	2003	2.34629713E-01	1519	2.22106066E-01

whose degree is $m-1$. We notice the this accuracy deteriorates also when m increases which is also normal since the number of unknowns increases with m and therefore a larger number of collocation points is necessary to have a good approximation.

Second case: $n_{1a} = 60$**,** $n_{1b} = 720$ **and** $Tol = 10^{-10}$.
In this second case, the number of boundary collocation points used for discretizing the boundary is taken to be $n_{1a} = 60$ and for the number of internal collocation points is $n_{1b} = 720$. For both algorithms CG and CGLS, we take tha same tolerance as the previous case, i.e. $Tol = 10^{-10}$ in the stopping criterion. The results are given in Tables 13 - 16.

We then observe from Tables 13 - 16 , an improvement in the results: the error is multiplied at least by 10^{-1} everywhere with always the same advantage for the CGLS method. On the other hand, as the number of collocation points is greater than in the first case, the size of

Table 11: Accuracy and convergence for $k = \sqrt{52}$, with $\rho(\theta) = 0.5$, $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.98601593E-01	3	9.98601593E-01
3	7	9.97777941E-01	7	9.97777941E-01
4	15	1.105533227	14	1.105533227
5	33	9.84224273E-12	32	1.54465433E-10
6	77	3.21395766E-09	80	1.87263828E-10
7	206	1.61872310E-08	172	2.30236344E-08
8	528	1.01149326E-06	429	2.03441111E-08
9	1739	8.36318678E-05	839	1.31616261E-02
10	2240	2.12334250E-02	1111	2.48436675E-02

Table 12: Accuracy and convergence for $k = \sqrt{100}$, with $\rho(\theta) = 0.5$, $\beta = 0.5$ and $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.98597910E-01	3	9.98597910E-01
3	7	9.97257349E-01	7	9.97257349E-01
4	15	1.116035107	15	1.116035107
5	33	2.07056017E-12	32	1.83247349E-11
6	72	2.67121160E-09	73	4.07846425E-10
7	214	2.59122498E-08	185	2.47142313E-08
8	530	4.06068221E-04	421	4.05984114E-04
9	1028	1.92944626E-03	996	1.35353567E-03
10	1504	2.16052265E-03	962	2.16362779E-03

Table 13: Accuracy and convergence for $k = \sqrt{15}$ with $\rho(\theta) = 0.5$, $\beta = 0.5$, $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.99390813E-01	3	9.99390813E-01
3	7	1.008242633	7	1.008242633
4	14	1.139608415	14	1.139608415
5	30	1.30314130E-10	30	1.64450714E-10
6	66	2.90750746E-10	62	1.60490278E-11
7	151	2.21294277E-09	134	1.23092324E-08
8	406	5.06551793E-08	328	1.89056150E-09
9	1260	9.76501466E-07	863	2.88519798E-07
10	4860	2.16859811E-05	2028	1.07061659E-02

the matrices is greater and consequently the number of iterations to reach the same tolerance becomes greater. The results confirm the superiority of CGLS which makes it possible to reduce this number of iterations. Other results, which we do not present here to avoid cluttering the presentation, confirm that the precision is improved for smaller tolerances. All these results allow us to conclude that the method approaches very well the solution of the Cauchy problem when the data are polynomial.

5.2 Non-polynomial example

The objective of this section is to test the effectiveness of the proposed method by considering data from non-polynomial functions. Our numerical results will relate to any number of waves. For this, we consider a small and a large wavenumber ($k = \sqrt{25.5}, \sqrt{100}$).

The domain considered here is bounded by $\rho(\theta) = 0.5$, the parameter β is taken to be $\beta = 0.5$,

Table 14: Accuracy and convergence for $k = \sqrt{25.5}$ with $\rho(\theta) = 0.5$, $\beta = 0.5$, $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.99351705E-01	3	9.99351705E-01
3	7	1.005554753	7	1.005554753
4	14	1.045136363	14	1.045136363
5	33	5.69473309E-11	32	4.18675265E-11
6	75	5.29480812E-11	64	3.80474385E-09
7	152	6.71573269E-09	141	6.12872602E-10
8	409	5.67460677E-08	323	1.17254779E-07
9	1182	5.13838465E-07	829	6.02598039E-06
10	3661	1.75109016E-02	2392	6.89820353E-05

Table 15: Accuracy and convergence for $k = \sqrt{52}$ with $\rho(\theta) = 0.5$, $\beta = 0.5$, $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.99336238E-01	3	9.99336238E-01
3	7	1.004535641	7	1.004535641
4	14	1.034679488	14	1.034679488
5	32	1.98986173E-11	32	1.41533893E-11
6	72	8.57320669E-10	70	9.14568007E-11
7	189	6.46039525E-09	167	1.01847030E-10
8	425	6.41411638E-08	369	2.21420609E-08
9	1223	4.27663885E-07	938	5.62839274E-09
10	2590	2.30318493E-03	1867	2.31617009E-03

Table 16: Accuracy and convergence for $k = \sqrt{100}$ with $\rho(\theta) = 0.5$, $\beta = 0.5$, $Tol = 10^{-10}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
2	3	9.99332685E-01	3	9.99332685E-01
3	7	1.004339928	7	1.004339928
4	14	1.036734123	14	1.036734123
5	31	1.10066460E-11	32	1.94768283E-12
6	70	2.32532558E-10	71	2.98955207E-11
7	175	6.99153646E-10	171	1.01161997E-10
8	481	1.77864896E-08	405	2.03411506E-07
9	1549	5.21810160E-07	1136	1.23789483E-06
10	1722	5.26458128E-04	1078	6.03715215E-04

the boundary conditions in (2)-(3) are computed from the exact solution $u_{\text{ex}} = \exp(x)\, cos(y)$. The number of boundary collocation points used for discretizing the boundary is taken to be $\mathrm{n}_{1a} = 25$, and for the number of internal collocation points is $n_{1b} = 500$.

For the stoping criterion, the tolerance is taken $Tol = 10^{-15}$. The results are presented in tables 17, 18.

In the tables 17, 18, we observe that the quality of the solution remains very good. The best precision is obtained for $m = 9$ for the two algorithms CG and $CGLS$ with approximately the same precision. But for the other values of m the results remain acceptable. This means that even if we don't know the value of m ensuring the best result, we just have to take a large enough m for a good approximation. Of course here the number of iterations necessary to reach convergence is much more important than for the preceding cases. The performance of the $CGLS$ method is always better than that of the CG method.

Table 17: Accuracy and and convergence for $k = \sqrt{25.5}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
7	165	1.11833391E-05	133	1.11833447E-05
8	377	2.56736172E-06	310	2.56734932E-06
9	1097	3.53983721E-07	797	3.52777601E-07
10	2963	6.06578639E-07	1701	5.85498895E-07
11	4240	4.48482263E-05	1937	7.08122664E-05
12	6214	6.46496231E-05	3311	6.92928386E-05
13	5624	7.38145570E-05	3233	7.35561628E-05
14	6491	6.96830002E-05	3829	6.94486181E-05

Table 18: Accuracy and and convergence for $k = \sqrt{100}$.

m	Iter	ErrorCG	Iter	ErrorCGLS
7	164	1.57319263E-06	163	1.57319247E-06
8	404	1.38286945E-07	375	1.38294007E-07
9	1068	1.60219640E-08	948	1.56409840E-08
10	1949	1.80955468E-07	1575	1.80425897E-07
11	1992	3.97797632E-07	1999	2.31756081E-07
12	2327	1.92686074E-07	1920	1.93815458E-07
13	2598	2.48302724E-07	2032	2.49309361E-07
14	2401	2.57874127E-07	1995	2.58632234E-07

5.3 Perturbation by a noise

In general, the data used for solving inverse problems are collected from certain measurements and therefore necessarily contain measurement errors. To simulate this situation, we must study the effect of introducing noise into the data on the quality of the obtained approximate solution. This type of test is important because the problems we solve are ill-posed. To do this, we use the following formula to perturb the given Cauchy data:

$$h(\theta) = u_{ex}(\rho(\theta), \theta) + \sigma * \omega$$

where ω is the standard deviation of measurement errors which is assumed to be the same for all measurements, and ω is the Gaussian distributed random error. σ represent the noise level, we use the values 0.001, 0.01, 0.05 and 0.1.

The effect of measurement errors on the quality of inverse solutions is discussed using the first example with polynomial data, for $\beta = 0.5$, $k = \sqrt{52}$.

In figures 1-4, we have represented the approximate solution on the boundary Γ_2, recovered using the CG and $CGLS$ algorithms in comparison with the exact solution. It can be seen in these figures that, when σ decreases, the obtained numerical approximation is closer to the exact solution. The numerical results obtained by the two algorithms are "equivalent". They are still a reasonably good approximation to the exact solution of the problem, even when the boundary data are perturbed by 10% relative random noise ($\sigma = 0.1$), since we solved an ill-conditioned problem. However, when solving a very ill-posed problem as we will see in the following sections the recovery of the approximate solution on the underspecified boundary Γ_2 then becomes not so good. The results remain unsatisfactory even when the two methods are used with regularization.

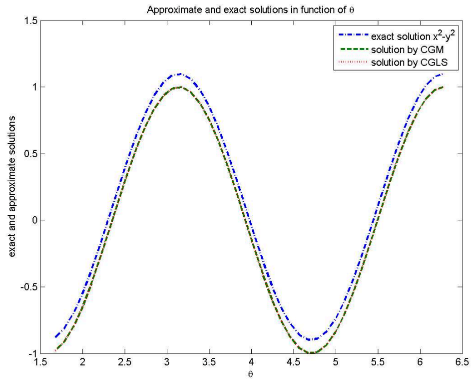

Figure 1: Exact and computed solutions with the anoise level $\sigma = 0.1$

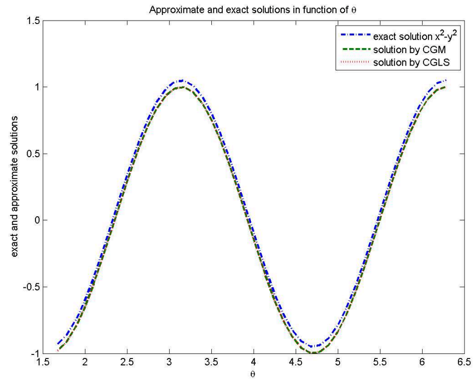

Figure 2: Exact and computed solutions with the anoise level $\sigma = 0.05$

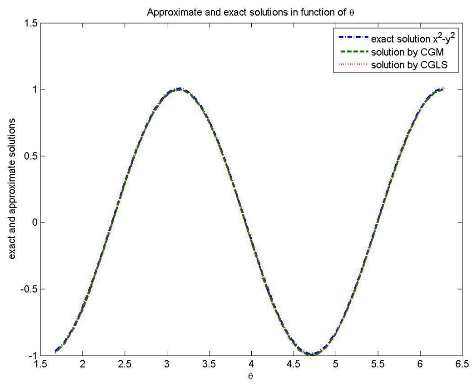

Figure 3: Exact and computed solutions with the anoise level $\sigma = 0.05$

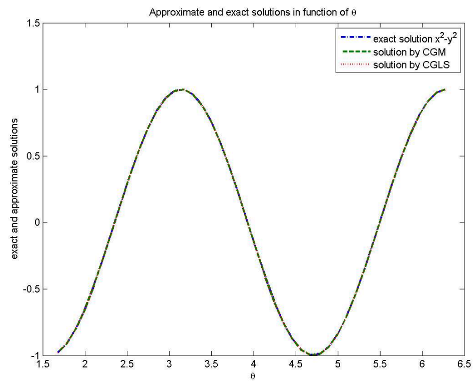

Figure 4: Exact and computed solutions with the anoise level $\sigma = 0.001$

5.4 Regularization and precondition

5.4.1 Effect of regularization and preconditioning

In the following, we examine the effect of regularization and preconditioning on the problem in the last non-polynomial example. We limit ourselves to the case where the resolution of the linear system is done with the CGLS algorithm. Applying regularization without preconditioning for different values of the parameter α does not improve either the precision or the number of iterations. In fact the regularization does not bring any improvement in accuracy and we only gain 0.5% (resp. 0.5%) of time for the case $k = \sqrt{25.5}$ (resp. $k = \sqrt{100}$). On the other hand, the application of preconditioning makes it possible to divide the number of iterations up to by three, see tables 19 - 20 .

The examples treated here are cases where the approximate solution is obtained with very good precision the application of a preconditioning acts as a convergence accelerator by reducing the number of iterations necessary for the CGLS algorithm to converge. In the following, we present results for cases with very ill-conditioned systems. In this cases, without preconditioning the method does not allow to obtain an acceptable solution.

Consider that the domain bounded by $\rho(\theta) = 0.5$. The parameter β defining Γ_1 and Γ_1, is taken to be $\beta = 0.5$ The boundary conditions in (2)-(3) are computed from the exact solution $u_{\text{ex}} = sin(p * x) * sinh(p * y) * (1/(p^2))$, for deferent values of p. The number of boundary collocation points used for discretizing the boundary is taken to be $\mathrm{n}_{1a} = 32$, and for the

Table 19: Results for $CGLS$, $k = \sqrt{25.5}$, $\rho(\theta) = 0.5$, $\beta = 0.5$, $n_{1a} = 25$, $n_{1b} = 500$ and $Tol = 10^{-15}$.

	Precondition	γ	α	Error by $CGLS$	Iter
without regularization & without precondition				3.52777601E-07	797
regularization & without precondition			10^{-12}	3.52987330E-07	793
regularization & preconditioning	right	0.1	10^{-12}	3.52711831E-07	395
	left	0.3	10^{-12}	2.04426056E-07	681
	two-sided	0.4	10^{-17}	2.67329650E-07	334

Table 20: Results for $CGLS$, $k = \sqrt{100}$, $\rho(\theta) = 0.5$, $\beta = 0.5$, $n_{1a} = 25$, $n_{1b} = 500$ and $Tol = 10^{-15}$.

	Precondition	γ	α	Error by $CGLS$	iter
without regularization & without precondition				1.56409840E-08	948
regularization & without precondition			10^{-12}	1.56414832E-08	937
regularization & preconditioning	right	0.2	10^{-14}	1.56293158E-08	349
	left	0.3	10^{-12}	2.04426056E-07	681
	two-sided	0.5	10^{-17}	3.06276812E-08	332

number of internal collocation points is $n_{1b} = 512$.

We start by examining the case where $p = 5$, and we take as tolerance $Tol = 10^{-12}$.

Table 21: Results for $CGLS$, $p = 5$ $k = \sqrt{15}$, $\beta = 0.5$, $n_{1a} = 32$, $n_{1b} = 512$ and $Tol = 10^{-12}$.

	Precondition	γ	α	Error	iteration
Without regularization, without preconditioning				0.151	25426
Regularization without preconditioning			10^{-20}	0.146	21182
Regularization & preconditioning	Left	0.8	10^{-13}	0.091	5279

For this example the best approximation is obtained for $m = 13$, the regularization without preconditioning brings a slight improvement about 16% reduction in the number of iterations but still no acceptable solution. In fact this problem is severely ill-conditioned with condition number $\kappa(A) = 1.74e + 012$. Fortunately, regularization with preconditioning has a positive effect on accuracy and execution time. for example, left-side preconditioning improves accuracy with a relative error $7.23E - 02$ and significantly reduces the number of iterations (number of iterations divided by 5).

Next, we consider the case $p = 8$ which is a case with more oscillation in the data. We take a smaller tolerance $Tol = 10^{-15}$ and we increase the number of collocation points $n_{1b} = 2048$.

We notice that despite the fact that we increased the number of collocation points and reduced the tolerance the method could not produce an acceptable solution even with regularization. This is due to the fact that the linear system is very badly conditioned. On the other hand, the use of proconditioners improves the results in terms of precision and execution time. For example, the the right-side preconditioning allows to build an approximation with a relative error of 0.079 in only 4091 iterations. That is to say an improvement in precision of 68% and a division of the number of iterations by 16.

Table 22: Results for $CGLS$, $p = 8$ $k = \sqrt{15}$, $\beta = 0.5$, $n_{1a} = 32$, $n_{1b} = 2048$ and $Tol = 10^{-15}$

	Precondition	γ	α	Error by $CGLS$	iter
Without regularization & without preconditioning				0.247	65934
Regularization & without preconditioning			10^{-11}	0.182	59745
Regularization & preconditioning	Right	0.3	10^{-6}	0.079	4091

6 Conclusion

We have solved the inverse Cauchy problem governed by a Helmholtz equation to recover unknown data on one part of the boundary from the overspecified Cauchy boundary conditions given on another inaccessible part. We transformed the inverse Cauchy problem to solve a direct problem, using polynomial expansion. A preconditioning strategy combined with a regularization was used in order to remedy the ill-conditioned character of the linear systems appearing in the determination of the expansion coefficients. Several numerical examples are presented to show that the method can overcome the very ill-posed property of the the aproximated inverse Cauchy problem. These numerical results also show that this meshless method makes it possible to obtain an acceptable solution even in the case of a large wavenumber, which is not the case for certain iterative methods used in the literature. However, it remains difficult to find an acceptable solution in certain cases where the data induce extremely ill-conditioned linear systems.

7 Acknowledgement

The authors are indebted to the referees of this paper for their most helpful comments and suggestions, which helped to improve the presentation greatly. The authors also warmly thank Professor A. Nachaoui of Nantes University for having proposed and co-supervised this work.

References

Aboud, F., Nachaoui, A. & Nachaoui, M. (2021). On the approximation of a Cauchy problem in a non-homogeneous medium, *J. Phys.: Conf. Ser., 1743*, 012003.

Berdawood, K.A., Nachaoui, A., Saeed, R., Nachaoui, M. & Aboud, F. (2022). An efficient $D - N$ alternating algorithm for solving an inverse problem for Helmholtz equation, *Discrete & Continuous Dynamical Systems-S, 5* (1), 57-78.

Berdawood, K. A., Nachaoui, A., Nachaoui, M. & Aboud, F. (2021). An effective relaxed alternating procedure for Cauchy problem connected with Helmholtz Equation, *Numer. Methods Partial Differential Equations* 1–27. https://doi.org/10.1002/num.22793

Berdawood, K.A., Nachaoui, A., Saeed, R., Nachaoui, M. & Aboud, F. (2020). An alternating procedure with dynamic relaxation for Cauchy problems governed by the modified Helmholtz equation, *Advanced Mathematical Models & Applications, 5* (1), 131-139.

Bergam, A. Chakib, A. Nachaoui,A & Nachaoui, M. (2019). Adaptive mesh techniques based on a posteriori error estimates for an inverse cauchy problem, *Appl. Math. Comput., 346*, 865–878.

Beskos DE (1997). Boundary element method in dynamic analysis: Part II, *ASME Appl Mech Rev, 50*, 986–1996.

Berntsson, F. Kozlov, V.A. Mpinganzima, L. & Turesson, B.O. (2017). Iterative Tikhonov regularization for the Cauchy problem for the Helmholtz equation, *Comput. Math. Appl., 73*(1), 163–172.

Berntsson, F. Kozlov, V.A., Mpinganzima, L. & Turesson, B.O., (2014). An alternating iterative procedure for the Cauchy problem for the Helmholtz equation, *Inverse Probl. Sci. Eng., 22*, 45–62.

Boulkhemair, A., Nachaoui, A. & Chakib, A. (2013), A shape optimization approach for a class of free boundary problems of Bernoulli type. *Appl. Math. 58,* (2), 205–221.

Chakib, A., Nachaoui, A., Nachaoui, M. & Ouaissa, H. (2019). On a fixed point study of an inverse problem governed by stokes equation, *Inverse Problems*, *35*(1), 015008.

Chen, J.T., Wong, F.C. (1998). Dual formulation of multiple reciprocity method for the acoustic mode of a cavity with a thin partition. *J Sound Vibration, 217*, 75–95.

Dvalishvili, Ph.; Nachaoui, A.; Nachaoui, M. & Tadumadze, T. (2017),. On the well-posedness of the Cauchy problem for a class of differential equations with distributed delay and the continuous initial condition. *Proc. Inst. Math. Mech. Natl. Acad. Sci. Azerb. 43* (1), 146–160.

Ellabib, A., Nachaoui, A. & Ousaadane A. (2021). Mathematical analysis and simulation of fixed point formulation of Cauchy problem in linear elasticity. *Mathematics and Computers in Simulation, 187*, 231-247.

Ellabib, A., Nachaoui, A. & Ousaadane A. (2022). Convergence study and regularizing property of a modified Robin-Robin method for the Cauchy problem in linear elasticity. *Inverse Problems, 38*(7), Paper No. 075007, 35.

Essaouini, M., Nachaoui, A. & El Hajji, S. (2004). Reconstruction of boundary data for a class of nonlinear inverse problems. *Journal of Inverse and Ill-Posed Problems, 12*(4), 369-385.

Fu, C.-L., Feng, X.-L & Qian, Z. (2009). The Fourier regularization for solving the Cauchy problem for the Helmholtz equation. *Appl. Numer. Math., 59*(10), 2625–2640.

Hadamard, J., (1953). *Lectures on Cauchy's problem in linear partial differential equations.* Dover Publications Dover Publications, New York.

Hansen, P.C., (1998). *Rank-deficient and discrete ill-posed problems: Numerical aspects of linear inversion.* SIAM, Philadelphia.

Harari, I., Barbone, P.E., Slavutin, M. & Shalom, R., (1998). Boundary infinite elements for the Helmholtz equation in exterior domains. *Int. J. Numer. Meth. Engng., 41,* 105–1131.

Hua, Q., Gu, Y., Qu, W., Chen, W. & Zhang, C. (2017). A meshless generalized finite difference method for inverse Cauchy problems associated with three-dimensional inhomogeneous Helmholtz-type equations. *Eng. Anal. Bound. Elem., 82*, 162–171.

Huang, C.H., Chen, W.C., (2000). A three-dimensional inverse forced convection problem in estimating surface heat flux by conjugate gradient method. *International Journal of Heat and Mass Transfer, 43*(17), 3171-3181.

Ihlenburg F., Babuska, I., (1997). Finite element solution of the helmholtz equation with high wave number part ii: The hp version of the fem. *SIAM J. Numer. Anal., 34*, 315–358.

Ihlenburg F., Babuska, I., (1995). Finite element solution of the helmholtz equation with high wave number part i: The h-version of the fem. *Comput. Math. Appl., 30*, 9–37.

Isakov, V. (2017). *Inverse problems for partial differential equations. Applied Mathematical Sciences,* 127, Springer, Cham.

Jourhmane, M., Nachaoui, A. (1999) An alternating method for an inverse Cauchy problem, Numerical Algorithms. *Numer. Algorithms, 21*(1-4), 247-260.

Juraev, D.A., Gasimov, Y.S. (2022). On the regularization Cauchy problem for matrix factorizations of the Helmholtz equation in a multidimensional bounded domain. Azerb. J. Math. 12 (1), 142–161.

Kraus, A.D., Aziz, A. & Welty, J. (2001). *An iterative method for solving the Cauchy problem for elliptic equations. Extended surface heat transfer.* John Wiley & Sons, New York.

Lavrentiev, M.M. (2013). *Some improperly posed problems of mathematical physics.* Springer Science & Business Media.

Liu, C.S., Wang, F. (2018) A meshless method for solving the nonlinear inverse Cauchy problem of elliptic type equation in a doubly-connected domain. *Comput. Math. Appl., 76*, 1837–1852.

Liu, J.-C., Wei, T. (2013) A quasi-reversibility regularization method for an inverse heat conduction problem without initial data, *Appl. Math. Comput., 219*(23),10866–10881.

Liu, C.S., Kuo, C.L. (2016) A multiple-scale Pascal polynomial triangle solving elliptic equations and inverse Cauchy problems. *Engineering Analysis with Boundary Elements, 62*, 35-43.

Mukanova, B. (2013). Numerical reconstruction of unknown boundary data in the Cauchy problem for Laplace's equation. *Inverse Probl. Sci. Eng., 21*(8), 1255–1267.

Nachaoui, M., Nachaoui, A. & Tadumadze, T. (2022) On the numerical approximation of some inverse problems governed by nonlinear delay differential equation. *RAIRO Oper. Res., 56*(3), 1553–1569.

Nachaoui, A., Aboud, F. & Nachaoui, M. (2021). Acceleration of the KMF algorithm convergence to solve the Cauchy problem for Poisson's equation. In: Nachaoui, A., Hakim, A., Laghrib, A. (eds) *Mathematical control and numerical applications,* 43–57, Springer Proc. Math. Stat., 372, Springer, Cham.

Nachaoui, A., Nachaoui, M., Chakib, A. & Hilal, M.A. (2021) Some novel numerical techniques for an inverse Cauchy problem. *J. Comput. Appl. Math., 381*, 113030.

Nachaoui, A., Salih, H.W. (2021). An analytical solution for the nonlinear inverse Cauchy problem. *Advanced Math. Models & Appl., 6* (3), 191-206.

Qian, Z., Feng, X. (2017). A fractional Tikhonov method for solving a Cauchy problem of Helmholtz equation. *Appl. Anal.*, *96*, 1656–1668.

Qian, A.L., Xiong, X.T., & Wu, Y.J. (2010). On a quasi-reversibility regularization method for a Cauchy problem of the Helmholtz equation. *Journal of Computational and Applied Mathematics, 233*(8), 1969-1979.

Ouaissa, H., Chakib, A., Nachaoui, A. & Nachaoui, M. (2022). On Numerical Approaches for Solving an Inverse Cauchy Stokes Problem. *Appl. Math. Optim., 85* (1), Paper No. 3, https://doi.org/10.1007/s00245-022-09833-8.

Rasheed, S.M., Nachaoui, A., Hama, M.F. & Jabbar, A.K. (2021) Regularized and preconditioned conjugate gradient like-methods methods for polynomial approximation of an inverse Cauchy problem, *Advanced Mathematical Models & Applications, 6*(2), 89-105.

Reddy, G.M.M., Nanda, P., Vynnycky, M., Cuminato, J.A. (2021). An adaptive boundary algorithm for the reconstruction of boundary and initial data using the method of fundamental solutions for the inverse Cauchy-Stefan problem. *Comput. Appl. Math., 40*(3), Paper No. 99.

Regińska, T., Regiński, K. (2006). Approximate solution of a Cauchy problem for the Helmholtz equation. *Inverse Problems, 22* (3), 975–989.

Yarmukhamedov, Sh., Yarmukhamedov, I. (2003). Cauchy problem for the Helmholtz equation. Ill-posed and non-classical problems of mathematical physics and analysis, 143–172, in it Inverse Ill-posed Probl. Ser., VSP, Utrecht.

Aster, R., Borchers, B. & Thurber C. (2012). *Parameter Estimation and Inverse Problems,* Academic Press.

Wang, F., Chen, Z., Li, P.-W., & Fan, C.-M. (2021). Localized singular boundary method for solving Laplace and Helmholtz equations in arbitrary 2D domains. *Eng. Anal. Bound. Elem., 129* , 82–92.

Wang, F., Gu, Y., Qu, W., & Zhang, C. (2020). Localized boundary knot method and its application to large-scale acoustic problems. *Comput. Methods Appl. Mech. Engrg., 361*, 112729, 30 pp.

Yang, F., Zhang, P. & Li, X.X. (2019). The truncation method for the Cauchy problem of the inhomogeneous Helmholtz equation. *Appl. Anal., 98*, 991-1004.