A combination of Kohn-Vogelius and DDM methods for a geometrical inverse problem

Slim Chaabane, Houssem Haddar, Rahma Jerbi

A combination of Kohn-Vogelius and DDM methods for a geometrical inverse problem

S. Chaabane[1], H. Haddar[2,*], R. Jerbi[1]

[1]Laboratory LAMHA-LR 11ES52, Department of Mathematics, Faculty of Sciences of Sfax, University of Sfax, BP 1171, Sfax 3000, Tunisia.
[2]INRIA, UMA, ENSTA Paris, Institut Polytechnique de Paris, 828, Bd des Maréchaux, 91762, Palaiseau Cedex, France.
*Corresponding Author

Abstract

We consider the inverse geometrical problem of identifying the discontinuity curve of an electrical conductivity from boundary measurements. This standard inverse problem is used as a model to introduce and study a combined inversion algorithm coupling a gradient descent on the Kohn-Vogelius cost functional with a domain decomposition method that includes the unknown curve in the domain partitioning. We prove the local convergence of the method in a simplified case and numerically show its efficiency for some two dimensional experiments.

1 Introduction

The goal of this paper is to introduce and study a combined iterative inversion method that uses incomplete solver for the direct problem at each iteration for the inverse problem. This type of approach has the advantage of reducing the iterative cost and speeding up the convergence rate. Many variants of these combinations have been proposed in the literature [19, 8, 18, 21, 16, 29, 6]. We develop here an algorithm in the vein of so-called one-shot iterative methods for optimization problems. We address two novel aspects for this type of methods. The first one is to use a combination of the Kohn-Vogelius energy functional and a non overlapping domain decomposition method as an iterative solver. The second one is that the inverse parameter is the geometry of an unknown object used as a part of the domain partitioning. We study all theses aspects in the framework of the inverse conductivity problem, where one would like to identify the discontinuity curve Σ of an electric conductivity from boundary measurements. This is a classic inverse problem that has many applications and has been extensively studied in the literature (we refer to [25, 7, 22, 33, 26, 30]). Our goal here is not to address issues specific to this inverse problem but to rather use it as a toy model to illustrate the feasibility of the combined inversion scheme and study convergence in some simplified configurations.

The Kohn-Vogelius cost functional has been used in the solution of various inverse geometrical problems [31, 12, 4, 23]. It has been specifically applied to inverse conductivity problem in [31, 4, 1, 2]. This functional seems to provide better stability and precision as compared with a classical least squares cost functional [10, 1]. From the theoretical point of view, it also

has the advantage of being differentiable with respect to the discontinuity curve under less restrictive smoothness assumptions [9, 11]. We shall review some results from the literature on this functional and study its local convergence for the inverse conductivity problem in the case of circular domains.

The implementation of this method requires the solution of two direct problems, one associated with Neumann data and the other one associated with Dirichlet data coming from the measurements. In order to evaluate these solutions we employ a non overlapping domain decomposition method. We employ the so-called Optimized Schwarz Method (OSM) where communications at the interfaces of the domains are enforced through Robin type boundary conditions [32, 27, 28, 15, 13]. We here study the case where the inverse problem unknown Σ is part of the partitioning used in the domain decomposition method. We propose a combined inverse algorithm where the Kohn-Vogelius cost functional is minimized using a gradient descent scheme. At each gradient step, the exact solutions of the Neumann and Dirichlet problems are approximated using only one or a few iterations of the OSM. Two difficulties arise in this coupling. The first one is that the domain partitioning changes after a gradient descent step which requires modifications in the OSM scheme. The second one is that the gradient cannot be evaluated exactly and therefore a choice has to be made. One should either evaluate the shape gradient with respect to incomplete exact solutions or propose an approximation of the gradient of the exact cost functional. The first option would require the introduction of two adjoint problems and therefore may render the method more costly. This is why we adopt the second approach, that indeed lead to an incorrect gradient at first iterations, but this gradient becomes closer to the exact one as the iterations number increases. Concerning the first issue related to OSM, we solve it by first rewriting the OSM as an iterative scheme on the impedance interface values (see (3.6)). These interface values are then transported on the updated geometry Σ using the shape gradient (see step 5 of Algorithm 2). We explicit this scheme in the case of star shaped interfaces, but the approach can be easily extended to other type or shape parametrizations. We study and prove local convergence of the resulting algorithm in the very simplified case where the geometry is circular and the inverse shape problem is the radius of the inner circle. We then numerically investigate the effectiveness and the accuracy of this algorithm in the case of star shaped domains. We show in particular that only one OSM iteration would achieve a converge rate similar to classical gradient (where the solutions are evaluated exactly at each iteration). Determining the optimal choice of OSM iteration number in order to have the best convergence rate is an open issue. Numerical tests suggests that this optimal choice is among the one using few number of OSM iterations. The current work addresses only the 2D problem. The extension to 3D problems would not bring any major difficulties with respect to the theoretical developments in the present work (considering the local convergence in the case of spheres). Also, the extension of the numerical algorithm can follow a similar procedure but indeed its implementation would be more involving. In fact, the true efficiency of the proposed combined inversion method can only be revealed for large scale problems such as 3D problems. This is the subject of an forthcoming work.

The article is organized as follows. We introduce the direct and inverse problem together with the Kohn-Vogelius cost functional in Section 2. The combined inversion algorithm is presented and studied in Section 3. We provide in particular a local convergence result in Section 3.3. Section 4 is dedicated to some numerical experiments for testing the efficiency of the combined algorithm and comparing with the classical one.

2 A model problem

Let Ω be a simply connected bounded domain of $\mathbb{R}^2$ with $\mathcal{C}^{1,\beta}$ boundary $\Gamma := \partial\Omega$, $\beta \in]0,1[$. We denote by $\sigma : \Omega \longrightarrow \mathbb{R}$ the electric conductivity of Ω assumed to be a piecewise constant function with a regular discontinuity surface Σ. More specifically, $\sigma \in \mathcal{S}_{ad}$ where

$$\mathcal{S}_{ad} := \left\{\sigma = \sigma_1\chi_{\Omega_1} + \sigma_2\chi_{\Omega_2};\ \sigma_1 > 0; \sigma_2 > 0;\ \Omega_1 \subset \Omega;\ \Sigma = \partial\Omega_1 \text{ a } \mathcal{C}^{4,\beta} \text{ Jordan curve; } \Omega_2 = \Omega \setminus \overline{\Omega_1}\right\},$$

χ_{Ω_i} denotes the characteristics function of a domain Ω_i.

We denote by $u \in H^1(\Omega)$ the electric potential which satisfies the following Neumann boundary value problem

$$(\mathcal{N}_\sigma) \begin{cases} -\mathrm{div}(\sigma\nabla u) = 0 & \text{in } \Omega, \\ \sigma\dfrac{\partial u}{\partial \nu} = \phi & \text{on } \Gamma, \end{cases}$$

where ν denotes the outward unit normal on Γ (see Figure 1) and $\phi \in L^2(\Gamma)$ the current flux through Γ that satisfies the compatibility condition: $\int_\Gamma \phi\, ds = 0$. To ensure uniqueness of the solution to problem $(\mathcal{N}_\sigma)$, we impose that

$$\int_\Sigma u\, ds = 0. \tag{2.1}$$

Indeed one can choose other conditions that ensures uniqueness of the solutions. The advantage of using (2.1) is that it can be naturally encoded in the domain decomposition scheme introduced later.

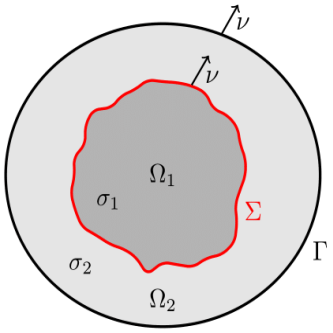


Figure 1: The domain Ω divided into two subdomains Ω_1 and Ω_2.

Remark 2.1. *All the following still hold if we assume that σ_1 and σ_2 are known bounded regular functions on all of Ω and are positive definite.*

2.1 The inverse problem and the Kohn-Vogelius functional

We consider in this section the inverse shape problem that consists in recovering the discontinuity interface $\overline{\Sigma}$ of an electrical conductivity $\overline{\sigma} \in S_{ad}$ from the knowledge of the flux ϕ together with the potential $f = u_{\overline{\sigma}_{|\Gamma}}$, where $u_{\overline{\sigma}}$ is the solution of $(\mathcal{N}_{\overline{\sigma}})$.

Remark 2.2. *Notice that, in general, a single couple (ϕ, f) is not sufficient to uniquely determine the unknown parameter $\overline{\sigma}$ [20]. However, two linearly independent pairs of measurements (ϕ_1, f_1), (ϕ_2, f_2) are sufficient to define the piecewise constant conductivity σ. Lipschitz stability results for the inverse geometrical problem using Dirichlet to Neumann maps have been*

established in [5] when the interface is polygonal and extended to polyhedral interfaces for 3D problems in [3].

Our adopted approach consists in transforming the inverse problem into an optimization one by constructing a cost function J modeling the energy gap between the solution of the direct problem and the solution of the following Dirichlet problem: $v \in H^1(\Omega)$, such that

$$(\mathcal{D}_\sigma) \begin{cases} -\mathrm{div}(\sigma \nabla v) = 0 & \text{in } \Omega, \\ v = f & \text{on } \Gamma. \end{cases}$$

More precisely, we define the Kohn-Vogelius cost function as

$$J(\sigma) := \int_\Omega \sigma \nabla(u_\sigma - v_\sigma) \cdot \nabla(u_\sigma - v_\sigma) dx$$

where $u_\sigma \in H^1(\Omega)$ the solution of the Neumann problem $(\mathcal{N}_\sigma)$ and $v_\sigma \in H^1(\Omega)$, the solution of the Dirichlet problem $(\mathcal{D}_\sigma)$. Indeed, the solution $\overline{\sigma}$ of the inverse problem is a zero of the non negative functional J (since $u_{\overline{\sigma}} = v_{\overline{\sigma}}$) and therefore is a minimizer of J.

To numerically minimize the function J, we shall use a gradient descent algorithm based on the shape derivative of J with respect to the singularity surface Σ of σ. The existence and expression of the shape derivative has been studied in [1] and we outline in the following the main related results.

Let $\sigma \in S_{ad}$ with $\Sigma = \overline{\Omega_1} \cap \overline{\Omega_2}$, $h > 0$ and $\zeta : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ a $\mathcal{C}^{4,\beta}$ vector field such that $\zeta = 0$ in a neighborhood of the boundary Γ. Then, there exists $h_0 > 0$, such that for all $h < h_0$, the mapping $\mathcal{F}_h = Id + h\zeta$ is a $\mathcal{C}^{4,\beta}$ diffeomorphism transforming the domain Ω into itself. We denote by $\Omega_{1,h} := \mathcal{F}_h(\Omega_1)$, $\Omega_{2,h} := \mathcal{F}_h(\Omega_2)$ and $\sigma_h = \sigma_1 \chi_{\Omega_{1,h}} + \sigma_2 \chi_{\Omega_{2,h}}$. Referring to [1, 2], the shape derivative of the cost function J is given by the following theorem.

Theorem 2.3. *The Kohn-Vogelius cost function J is differentiable with respect to the shape Σ and its derivative is given by:*

$$DJ(\sigma) \cdot \zeta = [\sigma] \int_\Sigma \left[\frac{1}{\sigma_1 \sigma_2} \left(\left| \sigma \frac{\partial v_\sigma}{\partial \nu} \right|^2 - \left| \sigma \frac{\partial u_\sigma}{\partial \nu} \right|^2 \right) + \left(|\nabla_\tau v_\sigma|^2 - |\nabla_\tau u_\sigma|^2 \right) \right] \zeta \cdot \nu \, ds,$$

where ζ is as defined above, $[\sigma] := \sigma_1 - \sigma_2$, ∇_τ denotes the tangential gradient and ν is the normal on Σ oriented to the exterior of Ω_1. The shape derivative is to be understood in the sense that

$$\frac{J(\sigma_h) - J(\sigma)}{h} = DJ(\sigma) \cdot \zeta + \epsilon(h) \quad \textit{where } \lim_{h \to 0} |\epsilon(h)| = 0.$$

2.2 The gradient algorithm in the case of starlike domains

As a preparatory step to the combined algorithm we explicit the gradient descent algorithm in the case of starlike interfaces Σ. Let $\mathcal{C}$ be the set of $\mathcal{C}^1$ piecewise Jordan curves of $\mathbb{R}^2$, $n \in \mathbb{N}^*$ and $R = (R_0, \ldots, R_{n-1}) \in (\mathbb{R}_+^*)^n$. For $i = 0, 1, \ldots, n$, we denote by $\theta_i := \dfrac{2\pi}{n} i$ and $M_i := (R_i \cos(\theta_i), R_i \sin(\theta_i))$. Let $\Sigma := \Sigma_R$ be the interface defined as the union of the n following arcs, $i = 0, \ldots, n-1$

$$S_i := \left\{ M = \hat{M}_i(t) := (\hat{R}_i(t) \cos(\hat{\theta}_i(t)),\ \hat{R}_i(t) \sin(\hat{\theta}_i(t)),\ t \in [0,1] \right\}, \tag{2.2}$$

where $\hat{R}_i(t) := tR_{i+1} + (1-t)R_i$ and $\hat{\theta}_i(t) := t\theta_{i+1} + (1-t)\theta_i$ (see Figure 2 for an illustration) and where for the notation convenience we have set $R_n = R_0$. We also set $S_n = S_0$. We define the interface operator $\mathcal{T}_n$ by:

$$\begin{array}{rccl} \mathcal{T}_n: & (\mathbb{R}_+^*)^n & \longrightarrow & \mathcal{C} \\ & R & \longmapsto & \Sigma_R = \mathcal{T}_n(R) := \displaystyle\bigcup_{i=0}^{n-1} S_i. \end{array} \tag{2.3}$$

For $R \in (\mathbb{R}_+^*)^n$ sufficiently small, we denote by $\Omega_1 := \Omega_{1,R}$ the interior domain limited by the interface Σ_R and by $\Omega_2 := \Omega_{2,R} = \Omega \setminus \overline{\Omega_{1,R}}$. The unknown of the inverse problem is $\overline{R} \in (\mathbb{R}_+^*)^n$ that corresponds with $\overline{\Sigma} = \Sigma_{\overline{R}}$.

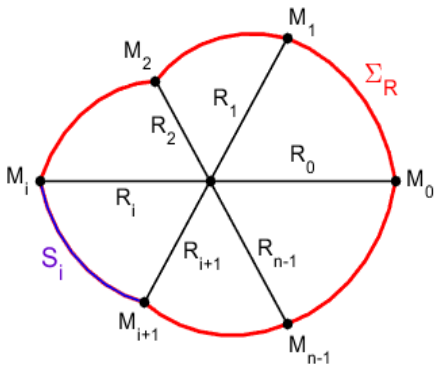


Figure 2: The subdomain $\Omega_{1,R}$.

Let us set $\sigma(R) := \sigma_1 \chi_{\Omega_{1,R}} + \sigma_2 \chi_{\Omega_{2,R}}$ and define the function $\mathcal{J}$ by

$$\begin{array}{rccl} \mathcal{J}: & (\mathbb{R}_+^*)^n & \longrightarrow & \mathbb{R} \\ & R & \longmapsto & \mathcal{J}(R) = J(\sigma(R)). \end{array} \tag{2.4}$$

The partial derivative $\dfrac{\partial \mathcal{J}}{\partial R_i}$ can be evaluated by applying Theorem 2.3 to a deformation field $\zeta = \zeta_i \nu$ on Σ_R where ζ_i is hat function defined by, for $i = 0, \ldots, n-1$,

$$\zeta_i(M) = t\chi_{\{M = \hat{M}_{i-1}(t) \in S_{i-1}\}} + (1-t)\chi_{\{M = \hat{M}_i(t) \in S_i\}}$$

with $S_{-1} = S_{n-1}$, $\hat{M}_i(t)$ is defined in (2.2) and ν is the outward normal vector to Ω_1. We then get, by using Theorem 2.3 that the derivative of the cost function $\mathcal{J}$ with respect to R_i is given by the following formula:

$$\frac{\partial \mathcal{J}}{\partial R_i}(R) = [\sigma(R)] \int_{\Sigma_R} \left[\frac{1}{\sigma_1 \sigma_2} \left(\left| \sigma(R) \frac{\partial v_{\sigma(R)}}{\partial \nu} \right|^2 - \left| \sigma(R) \frac{\partial u_{\sigma(R)}}{\partial \nu} \right|^2 \right) + \left(|\nabla_\tau v_{\sigma(R)}|^2 - |\nabla_\tau u_{\sigma(R)}|^2 \right) \right] \zeta_i \, ds. \tag{2.5}$$

A gradient descent scheme to solve the inverse problem is summarized in Algorithm 1.

Algorithm 1: Gradient descent algorithm with exact direct solver

- Fix the number of parameters $n \in \mathbb{N}^*$ that serve to define the starlike interface.
- Consider an initial guess $R^0 \in (\mathbb{R}_+^*)^n$, the initial interface $\Sigma = \Sigma_{R^0} = \mathcal{T}_n(R^0)$ and the corresponding conductivity $\sigma(R^0) = \sigma_1 \chi_{\Omega_{1,R^0}} + \sigma_2 \chi_{\Omega_{2,R^0}}$ as defined above.
- $k = 0$.

repeat *until* $k \leq$ *maximum number of iterations*

- Use a direct solver to calculate $u_{\sigma(R^k)}$ and $v_{\sigma(R^k)}$, the respective solutions of $(\mathcal{N}_{\sigma(R^k)})$ and $(\mathcal{D}_{\sigma(R^k)})$.
- Calculate $\dfrac{\partial \mathcal{J}}{\partial R_i}(R^k)$, for $i = 0, \ldots, n-1$ using formula (2.5).
- Update $\Sigma = \mathcal{T}_n(R^{k+1})$ with

$$R_i^{k+1} := R_i^k - \tau \frac{\partial \mathcal{J}}{\partial R_i}(R^k),\ i = 0, \ldots, n-1,$$

  where $\tau > 0$ is chosen sufficiently small (a step adaptation can be incorporated here).
- $R^k = R^{k+1}$.
- $k = k + 1$.

end

As a first step for the analysis of the combined inverse scheme introduced later, we study the convergence of this algorithm in the particular case where both of the domains Ω and Ω_1 are circular.

2.3 Local convergence analysis of the gradient descent algorithm

We study here the convergence of Algorithm 1 in the case $n = 1$, i.e $\Omega_{1,R}$ is an open disk of center $(0,0)$ and radius $R > 0$. We also choose the domain Ω to be the open disk of center $(0,0)$ and radius $R_2 > 0$. In this case the interface $\Sigma = \Sigma_R$ coincides with the circle of center $(0,0)$ and radius R (note that $S_{-1} = S_0 = S_1$). The unknown of the inverse problem is $\overline{R}$ that corresponds with $\overline{\Sigma} = \Sigma_{\overline{R}}$. We impose the current flux $\phi(\theta) = m\cos(m\theta)$ or $\phi(\theta) = m\sin(m\theta)$, $\theta \in [0, 2\pi]$ and $m \in \mathbb{N}^*$. The solution of the direct problem $(\mathcal{N}_{\sigma(R)})$ then can be explicitly expressed as

$$u_{\sigma(R)}(r,\theta) = \begin{cases} u_1(r,\theta) = \alpha_N\, r^m \dfrac{\phi(\theta)}{m} & \text{in } \Omega_{1,R}, \\[2ex] u_2(r,\theta) = \left(\beta_N\, r^m + \dfrac{\gamma_N}{r^m}\right)\dfrac{\phi(\theta)}{m} & \text{in } \Omega_{2,R}, \end{cases}$$

where:

$$\begin{cases} \alpha_N := \alpha_N(R) = \dfrac{2R_2^{m+1}}{\sigma_1(R_2^{2m} + R^{2m}) + \sigma_2(R_2^{2m} - R^{2m})}, \\[2ex] \beta_N := \beta_N(R) = \dfrac{(\sigma_2 + \sigma_1)R_2^{m+1}}{\sigma_2\left[\sigma_1(R_2^{2m} + R^{2m}) + \sigma_2(R_2^{2m} - R^{2m})\right]}, \\[2ex] \gamma_N := \gamma_N(R) = \dfrac{(\sigma_2 - \sigma_1)R_2^{m+1}R^{2m}}{\sigma_2\left[\sigma_1(R_2^{2m} + R^{2m}) + \sigma_2(R_2^{2m} - R^{2m})\right]}. \end{cases} \tag{2.6}$$

The measurement f is then given by:

$$f(\theta) = C_f \frac{\phi(\theta)}{m}$$

where

$$C_f := \beta_N(\overline{R})\ R_2^m + \frac{\gamma_N(\overline{R})}{R_2^m} = \frac{R_2}{\sigma_2}\frac{[\sigma_1(R_2^{2m} - \overline{R}^{2m}) + \sigma_2(R_2^{2m} + \overline{R}^{2m})]}{[\sigma_1(R_2^{2m} + \overline{R}^{2m}) + \sigma_2(R_2^{2m} - \overline{R}^{2m})]}. \tag{2.7}$$

Consequently, the solution of the Dirichlet problem $(\mathcal{D}_{\sigma(R)})$ is

$$v_{\sigma(R)}(r,\theta) = \begin{cases} v_1(r,\theta) = \alpha_D\ r^m \dfrac{\phi(\theta)}{m} & \text{in } \Omega_{1,R}, \\ \\ v_2(r,\theta) = \left(\beta_D\ r^m + \dfrac{\gamma_D}{r^m}\right)\dfrac{\phi(\theta)}{m} & \text{in } \Omega_{2,R}, \end{cases}$$

where:

$$\begin{cases} \alpha_D := \alpha_D(R) = \dfrac{2\sigma_2 R_2^m C_f}{\sigma_1(R_2^{2m} - R^{2m}) + \sigma_2(R_2^{2m} + R^{2m})}, \\ \\ \beta_D := \beta_D(R) = \dfrac{(\sigma_2 + \sigma_1) R_2^m C_f}{\sigma_1(R_2^{2m} - R^{2m}) + \sigma_2(R_2^{2m} + R^{2m})}, \\ \\ \gamma_D := \gamma_D(R) = \dfrac{(\sigma_2 - \sigma_1) R_2^m R^{2m} C_f}{\sigma_1(R_2^{2m} - R^{2m}) + \sigma_2(R_2^{2m} + R^{2m})}. \end{cases} \tag{2.8}$$

The function $\mathcal{J}$ depends here only on one variable R and its derivative is given by (2.5) for $i = 0$ with $\zeta_0 = 1$, namely,

$$\begin{aligned} \mathcal{J}'(R) &= [\sigma(R)] \int_0^{2\pi} \left[\frac{1}{\sigma_1\sigma_2} \left(\left| \sigma(R) \frac{\partial v_{\sigma(R)}}{\partial \nu} \right|^2 - \left| \sigma(R) \frac{\partial u_{\sigma(R)}}{\partial \nu} \right|^2 \right) + \left(|\nabla_\tau v_{\sigma(R)}|^2 - |\nabla_\tau u_{\sigma(R)}|^2 \right) \right] R\, d\theta \\ &= \frac{\pi m^2 (\sigma_1^2 - \sigma_2^2)}{\sigma_2} R^{2m-1} (\alpha_D^2(R) - \alpha_N^2(R)). \end{aligned}$$

Let us consider now the iterative sequence R^k obtained by the gradient algorithm:

$$R^{k+1} = R^k - \tau \mathcal{J}'(R^k)$$

where $\tau > 0$ denotes the descent step of the gradient algorithm.

Definition 2.4. *We say that the sequence R^k is locally convergent, if there exists $\varepsilon > 0$, such that, for every $R^0 \in]\overline{R} - \varepsilon, \overline{R} + \varepsilon[$, the sequence R^k converges to $\overline{R}$.*

We then have the following proposition.

Proposition 2.1. *Assume that $\sigma_1 \neq \sigma_2$ and let $\phi(\theta) = m\cos(m\theta)$ or $\phi(\theta) = m\sin(m\theta)$, $m \in \mathbb{N}^*$. Then $\mathcal{J}''(\overline{R}) > 0$ and the sequence R^k is locally convergent if and only if the descent step $\tau < 2/\mathcal{J}''(\overline{R})$.*

Proof. The function $\mathcal{J}$ is twice differentiable on $]0, R_2[$ and we have

$$\mathcal{J}''(R) = F_1'(R)F_2(R) + F_1(R)F_2'(R)$$

where

$$F_1(R) := \frac{\pi m^2(\sigma_1^2 - \sigma_2^2)}{\sigma_2} R^{2m-1} \text{ and } F_2(R) := \alpha_D^2(R) - \alpha_N^2(R).$$

From the fact $\alpha_N(\overline{R}) = \alpha_D(\overline{R})$, we deduce that

$$\mathcal{J}''(\overline{R}) = F_1(\overline{R})F_2'(\overline{R}) = 2F_1(\overline{R})\alpha_N(\overline{R})\left(\alpha_D'(\overline{R}) - \alpha_N'(\overline{R})\right),$$

where:

$$\alpha_N'(\overline{R}) = \frac{4(\sigma_2 - \sigma_1)mR_2^{m+1}\overline{R}^{2m-1}}{\left[\sigma_1(R_2^{2m} + \overline{R}^{2m}) + \sigma_2(R_2^{2m} - \overline{R}^{2m})\right]^2}$$

and

$$\alpha_D'(\overline{R}) = \frac{4(\sigma_1 - \sigma_2)m\sigma_2 C_f R_2^m \overline{R}^{2m-1}}{\left[\sigma_1(R_2^{2m} - \overline{R}^{2m}) + \sigma_2(R_2^{2m} + \overline{R}^{2m})\right]^2}$$

By using (2.7), we obtain

$$\alpha_D'(\overline{R}) = \frac{4(\sigma_1 - \sigma_2)mR_2^{m+1}\overline{R}^{2m-1}}{\left[\sigma_1(R_2^{2m} - \overline{R}^{2m}) + \sigma_2(R_2^{2m} + \overline{R}^{2m})\right]\left[\sigma_1(R_2^{2m} + \overline{R}^{2m}) + \sigma_2(R_2^{2m} - \overline{R}^{2m})\right]}$$

consequently

$$\begin{aligned} F_2'(\overline{R}) &= 2\,\alpha_N(\overline{R})\left(\alpha_D'(\overline{R}) - \alpha_N'(\overline{R})\right) \\ &= \frac{32m(\sigma_1^2 - \sigma_2^2)R_2^{4m+2}\overline{R}^{2m-1}}{\left[\sigma_1(R_2^{2m} - \overline{R}^{2m}) + \sigma_2(R_2^{2m} + \overline{R}^{2m})\right]\left[\sigma_1(R_2^{2m} + \overline{R}^{2m}) + \sigma_2(R_2^{2m} - \overline{R}^{2m})\right]^3}, \end{aligned}$$

and then

$$\mathcal{J}''(\overline{R}) = \frac{32\pi m^3(\sigma_1^2 - \sigma_2^2)^2 R_2^{4m+2}\overline{R}^{4m-2}}{\sigma_2\left[\sigma_1(R_2^{2m} - \overline{R}^{2m}) + \sigma_2(R_2^{2m} + \overline{R}^{2m})\right]\left[\sigma_1(R_2^{2m} + \overline{R}^{2m}) + \sigma_2(R_2^{2m} - \overline{R}^{2m})\right]^3}.$$

The iterations for R^k can be written as $R^{k+1} = g(R^k)$, where

$$g(R) = R - \tau\mathcal{J}'(R).$$

Then, the sequence R^k converges locally if and only if $|g'(\overline{R})| = |1 - \mathcal{J}''(\overline{R})| < 1$. From the condition: $\sigma_1 \neq \sigma_2$, we deduce that $\mathcal{J}''(\overline{R}) > 0$, and the sequence R^k converges locally if and only if $0 < \tau < \dfrac{2}{\mathcal{J}''(\overline{R})}$. □

3 A combined inversion method

We develop in this section some inversion algorithms combining the previous gradient algorithm with a Domain Decomposition Method (DDM) that respects the partitioning of the domain Ω into $\Omega_1 \cup \Omega_2 \cup \Sigma$. The main idea consists in approximating the direct problems $(\mathcal{N}_{\sigma(R^k)})$ and $(\mathcal{D}_{\sigma(R^k)})$ (at each iteration k of Algorithm 1) using only one or a few DDM steps. Prior to giving the details of the resulting algorithm, we shall present first the so-called Optimized Schwarz Method (OSM) that we use as DDM.

3.1 Multidomain formulation using OSM

3.1.1 The case of the direct problem $(\mathcal{N}_\sigma)$

Let us present here the OSM for solving the direct problem $(\mathcal{N}_\sigma)$. Let $u_i := u_{|\Omega_i}$ be the restriction of u to Ω_i, $i = 1, 2$. Then problem $(\mathcal{N}_\sigma)$ can be reformulated as an equivalent multidomain problem consisting of the following subdomain problems

$$\begin{aligned} -\sigma_i \Delta u_i &= 0 \quad \text{in } \Omega_i, \\ \sigma_2 \frac{\partial u_2}{\partial \nu} &= \phi \quad \text{on } \Gamma, \end{aligned}$$

together with the transmission conditions on the interface Σ

$$u_1 = u_2 \qquad \text{on } \Sigma, \tag{3.1}$$

$$\sigma_1 \frac{\partial u_1}{\partial \nu} = \sigma_2 \frac{\partial u_2}{\partial \nu} \quad \text{on } \Sigma. \tag{3.2}$$

Alternatively and equivalently, one may impose the Robin transmission conditions,

$$\sigma_i \frac{\partial u_i}{\partial \nu} \pm \alpha u_i = \sigma_j \frac{\partial u_j}{\partial \nu} \pm \alpha u_j \quad \text{on } \Sigma \ \text{ for all } i = 1, 2 \text{ and } j \neq i, \tag{3.3}$$

where $\alpha > 0$ is a fixed parameter that may be optimized to improve the convergence rate of the iterative domain decomposition method (see [32, 27, 28, 15, 13]). The resulting method is referred to as the Optimized Schwarz Method, which can be described as follows. Given arbitrary initial guesses $(u_i^0)_{1 \leq i \leq 2} \in H^2(\Omega_i)$, we inductively build the sequences $u_i^\ell \in H^1(\Omega_i), i = 1, 2$, by solving (in parallel) for all $\ell \geq 0$

$$\begin{aligned} -\sigma_i \Delta u_i^{\ell+1} &= 0 && \text{in } \Omega_i, \\ \sigma_2 \frac{\partial u_2^{\ell+1}}{\partial \nu} &= \phi && \text{on } \Gamma, \\ \sigma_1 \frac{\partial u_1^{\ell+1}}{\partial \nu} + \alpha u_1^{\ell+1} &= \sigma_2 \frac{\partial u_2^{\ell}}{\partial \nu} + \alpha u_2^{\ell} && \text{on } \Sigma, \\ \sigma_2 \frac{\partial u_2^{\ell+1}}{\partial \nu} - \alpha u_2^{\ell+1} &= \sigma_1 \frac{\partial u_1^{\ell}}{\partial \nu} - \alpha u_1^{\ell} && \text{on } \Sigma. \end{aligned} \tag{3.4}$$

A direct use of (3.3) would require a numerical evaluation of the normal derivatives along the interfaces Σ in order to compute the right-hand sides in the transmission conditions of (3.4). This can be avoided by renaming the problematic quantities

$$\lambda_{i,N}^\ell := \sigma_j \frac{\partial u_j^\ell}{\partial \nu} \pm \alpha u_j^\ell \quad \text{respectively for } \ i = 1, 2 \text{ and } j = 1, 2, \ j \neq i. \tag{3.5}$$

where $\lambda_{i,N}^{\ell}$ is the information coming from the neighboring subdomain Ω_j ($j \neq i$) at step ℓ of the algorithm. One can easily verify that

$$\begin{cases} \lambda_{1,N}^{\ell+1} = \sigma_2 \dfrac{\partial u_2^{\ell+1}}{\partial \nu} + \alpha u_2^{\ell+1} = \sigma_2 \dfrac{\partial u_2^{\ell+1}}{\partial \nu} - \alpha u_2^{\ell+1} + 2\alpha\, u_2^{\ell+1}, \\ \\ \lambda_{2,N}^{\ell+1} = \sigma_1 \dfrac{\partial u_1^{\ell+1}}{\partial \nu} - \alpha u_1^{\ell+1} = \sigma_1 \dfrac{\partial u_1^{\ell+1}}{\partial \nu} + \alpha u_1^{\ell+1} - 2\alpha u_1^{\ell+1}. \end{cases}$$

Therefore, the parameters $\lambda_{i,N}^{\ell}$ satisfy the induction

$$\begin{cases} \lambda_{1,N}^{\ell+1} &= \lambda_{2,N}^{\ell} + 2\alpha\, u_2^{\ell+1}, \\ \lambda_{2,N}^{\ell+1} &= \lambda_{1,N}^{\ell} - 2\alpha\, u_1^{\ell+1} \end{cases} \tag{3.6}$$

which can replace (3.5). We now can rewrite the iterative algorithm as follows. Given an initial guess $(\lambda_{i,N}^0)_{1\leq i\leq 2} \in L^2(\Sigma)$ solve for each iteration $\ell \in \mathbb{N}$ the two following problems:

$$\begin{aligned} -\sigma_i \Delta u_i^{\ell+1} &= 0 && \text{in } \Omega_i, \\ \sigma_2 \frac{\partial u_2^{\ell+1}}{\partial \nu} &= \phi && \text{on } \Gamma, \\ \sigma_1 \frac{\partial u_1^{\ell+1}}{\partial \nu} + \alpha u_1^{\ell+1} &= \lambda_{1,N}^{\ell} && \text{on } \Sigma, \\ \sigma_2 \frac{\partial u_2^{\ell+1}}{\partial \nu} - \alpha u_2^{\ell+1} &= \lambda_{2,N}^{\ell} && \text{on } \Sigma, \end{aligned} \tag{3.7}$$

where the boundary values $\lambda_{i,N}^{\ell}$, $i = 1, 2$ are updated using (3.6). For the convergence analysis of this algorithm, we refer to [32, 17, 14]. Let us observe that condition (2.1) is ensured at convergence as soon as it is verified by the initial guess $(\lambda_{i,N}^0)_{1\leq i\leq 2}$. This is what we summarize in the following lemma.

Lemma 3.1. *Let $(\lambda_{i,N}^0)_{1\leq i\leq 2} \in L^2(\Sigma)$, such that $\displaystyle\int_\Sigma \lambda_{i,N}^0\, ds = 0$; $i = 1, 2$, then for $\ell \in \mathbb{N}^*$, we have: $\displaystyle\int_\Sigma \lambda_{i,N}^{\ell}\, ds = 0$, and $\displaystyle\int_\Sigma u_i^{\ell}\, ds = 0$.*

Proof. We prove this Lemma by induction. By integrating equations (3.7) for $\ell = 0$ in Ω_i against a constant function in Ω_i we obtain

$$\begin{cases} \displaystyle\int_\Sigma u_1^1\, ds = -\frac{1}{\alpha}\left[\int_\Sigma \lambda_{1,N}^0\, ds\right] = 0 \\ \displaystyle\int_\Sigma u_2^1\, ds = \frac{1}{\alpha}\left[-\int_\Sigma \lambda_{2,N}^0\, ds + \int_\Gamma \phi\, ds\right] = 0, \end{cases}$$

which proves the statement for $\ell = 1$. Assume that

$$\int_\Sigma \lambda_{i,N}^{\ell}\, ds = \int_\Sigma u_i^{\ell}\, ds = 0, \text{ for } i = 1, 2.$$

By integrating equations (3.7) in Ω_i against a constant function in Ω_i we obtain

$$\begin{cases} \displaystyle\int_\Sigma u_1^{\ell+1}\, ds = -\frac{1}{\alpha}\left[\int_\Sigma \lambda_{1,N}^{\ell}\, ds\right] = 0, \\ \displaystyle\int_\Sigma u_2^{\ell+1}\, ds = \frac{1}{\alpha}\left[-\int_\Sigma \lambda_{2,N}^{\ell}\, ds + \int_\Gamma \phi\, ds\right] = 0 \end{cases}$$

and by using (3.6), we obtain

$$\int_\Sigma \lambda_{1,N}^{\ell+1}\, ds = \int_\Sigma \lambda_{2,N}^{\ell+1}\, ds = 0,$$

which prove the lemma. □

We study in the sequel the convergence rate of OSM in the case of circular interfaces, which will be useful for the convergence analysis of the combined inversion algorithm.

3.1.2 Convergence rate for OSM in circular domains

We consider again the particular case where the domain Ω_1 is the open disk of center $(0,0)$ and radius $R>0$ of $\mathbb{R}^2$ and an Ω_2 is the annulus domain $R<|x|<R_2$. The interface Σ coincides with the circle of center $(0,0)$ and radius R. We set for $m>0$,

$$k_m^\alpha(R) := \frac{\sigma_1 m - \alpha R}{\sigma_1 m + \alpha R}, \tag{3.8}$$

$$p_m^\alpha(R) := \frac{\sigma_2 m(R_2^{2m} - R^{2m}) - \alpha R(R_2^{2m} + R^{2m})}{\sigma_2 m(R_2^{2m} - R^{2m}) + \alpha R(R_2^{2m} + R^{2m})}. \tag{3.9}$$

Then, we have the following proposition:

Proposition 3.2. *For $\phi(\theta) = m\cos(m\theta)$ or $\phi(\theta) = m\sin(m\theta)$, $\theta \in [0,2\pi]$ and $m \in \mathbb{N}^*$, the OSM (3.6)-(3.7), initialized with $\lambda_{i,N}^0 = 0$, $i=1,2$, geometrically converges for all $\alpha>0$ and the convergence rate is given by the spectral radius*

$$\rho_m^\alpha(R) := \sqrt{|\, k_m^\alpha(R) p_m^\alpha(R)\,|} < 1.$$

Proof. For $\phi(\theta) = m\cos(m\theta)$ or $\phi(\theta) = m\sin(m\theta)$, $\theta \in [0,2\pi]$ and $m \in \mathbb{N}^*$, the solutions of the problem (3.7) can be written as:

$$u^{\ell+1}(r,\theta) = \begin{cases} u_1^{\ell+1}(r,\theta) = \alpha_N^{\ell+1}\, r^m \dfrac{\phi(\theta)}{m} & \text{in } \Omega_1, \\[2ex] u_2^{\ell+1}(r,\theta) = \left(\beta_N^{\ell+1}\, r^m + \dfrac{\gamma_N^{\ell+1}}{r^m}\right)\dfrac{\phi(\theta)}{m} & \text{in } \Omega_2, \end{cases}$$

where the constants $\alpha_N^{\ell+1}$, $\beta_N^{\ell+1}$ and $\gamma_N^{\ell+1}$ are determined by the following equations:

$$\begin{cases} \sigma_2 \dfrac{\partial u_2^{\ell+1}}{\partial \nu} & = \phi(\theta) \quad \text{on } \Gamma, \\[2ex] \sigma_1 \dfrac{\partial u_1^{\ell+1}}{\partial \nu} + \alpha u_1^{\ell+1} & = \lambda_{1,N}^\ell \quad \text{on } \Sigma, \\[2ex] \sigma_2 \dfrac{\partial u_2^{\ell+1}}{\partial \nu} - \alpha u_2^{\ell+1} & = \lambda_{2,N}^\ell \quad \text{on } \Sigma. \end{cases} \tag{3.10}$$

For $i=1,2$, we denote by $\hat{\lambda}_{i,N}^\ell = \lambda_{i,N}^\ell \dfrac{m}{\phi(\theta)}$. Then, from the second equation of (3.10), we obtain:

$$u_1^{\ell+1}(r,\theta) = \frac{\hat{\lambda}_{1,N}^\ell}{R^{m-1}(m\sigma_1 + \alpha R)} r^m \frac{\phi(\theta)}{m} \tag{3.11}$$

and from the first and the third equations of (3.10), we obtain:

$$u_2^{\ell+1}(r,\theta) = \left[\frac{R_2^{m+1} R^{m+1} \left((\sigma_2 m - \alpha R) R^{m-1} - \sigma_2 \hat{\lambda}_{2,N}^{\ell} R_2^{m-1} \right)}{\sigma_2 \left(\sigma_2 m (R_2^{2m} - R^{2m}) + \alpha R (R_2^{2m} + R^{2m}) \right)} \left(\frac{r^m}{R_2^{2m}} + \frac{1}{r^m} \right) + \frac{r^m}{\sigma_2 R_2^{m-1}} \right] \frac{\phi(\theta)}{m}.$$

From (3.6) and using the fact that $\lambda_{i,N}^{\ell+1} = \hat{\lambda}_{i,N}^{\ell+1} \dfrac{\phi(\theta)}{m}$, we obtain:

$$\hat{\lambda}_{2,N}^{\ell+1} = k_m^{\alpha}(R)\ \hat{\lambda}_{1,N}^{\ell}.$$

In the same way we get:

$$\hat{\lambda}_{1,N}^{\ell+1} = p_m^{\alpha}(R)\ \hat{\lambda}_{2,N}^{\ell} + \eta_m^{\alpha}(R),$$

where

$$\eta_m^{\alpha}(R) := \frac{4m\alpha R^m R_2^{m+1}}{\sigma_2 m (R_2^{2m} - R^{2m}) + \alpha R (R_2^{2m} + R^{2m})}. \tag{3.12}$$

Therefore, the DDM iterations can be written as:

$$\begin{pmatrix} \hat{\lambda}_{1,N}^{\ell+1} \\ \\ \hat{\lambda}_{2,N}^{\ell+1} \end{pmatrix} = \begin{pmatrix} 0 & p_m^{\alpha}(R) \\ \\ k_m^{\alpha}(R) & 0 \end{pmatrix} \begin{pmatrix} \hat{\lambda}_{1,N}^{\ell} \\ \\ \hat{\lambda}_{2,N}^{\ell} \end{pmatrix} + \begin{pmatrix} \eta_m^{\alpha}(R) \\ \\ 0 \end{pmatrix}. \tag{3.13}$$

This induction converges if and only if $\rho_m^{\alpha}(R) = \sqrt{\mid k_m^{\alpha}(R) p_m^{\alpha}(R) \mid} < 1$, which is always true for $\alpha > 0$, $m > 0$ and $0 < R < R_2$. □

3.1.3 The case of the Dirichlet problem $(\mathcal{D}_\sigma)$

Similarly to above, we also apply the OSM for solving the Dirichlet problem $(\mathcal{D}_\sigma)$. Let $v_i := v_{|\Omega_i}$ be the restriction of v to Ω_i, $i = 1, 2$. Then problem $(\mathcal{D}_\sigma)$ can be reformulated as an equivalent multidomain problem consisting of the following subdomain problems

$$\begin{aligned} -\sigma_i \Delta v_i &= 0 \quad \text{in } \Omega_i, \\ v_2 &= f \quad \text{on } \Gamma, \end{aligned}$$

together with the Robin transmission conditions:

$$\sigma_i \frac{\partial v_i}{\partial \nu} \pm \alpha v_i = \sigma_j \frac{\partial v_j}{\partial \nu} \pm \alpha v_j \quad \text{on } \Sigma \ \text{ for all } i = 1, 2 \text{ and } j \neq i. \tag{3.14}$$

The OSM for this problem can be then formulated as: Given an initial guess $(\lambda_{i,D}^0)_{1 \leq i \leq 2} \in L^2(\Sigma)$ solve for each iteration $\ell \in \mathbb{N}$ the two following problems:

$$\begin{aligned} -\sigma_i \Delta v_i^{\ell+1} &= 0 && \text{in } \Omega_i, \\ v_2^{\ell+1} &= f && \text{on } \Gamma, \\ \sigma_1 \frac{\partial v_1^{\ell+1}}{\partial \nu} + \alpha v_1^{\ell+1} &= \lambda_{1,D}^{\ell} && \text{on } \Sigma, \\ \sigma_2 \frac{\partial v_2^{\ell+1}}{\partial \nu} - \alpha v_2^{\ell+1} &= \lambda_{2,D}^{\ell} && \text{on } \Sigma, \end{aligned} \tag{3.15}$$

where the parameters $\lambda_{i,D}^{\ell+1}$, $i = 1, 2$, verify the induction

$$\left\{ \begin{array}{rcl} \lambda_{1,D}^{\ell+1} & = & \lambda_{2,D}^{\ell} + 2\alpha\ v_2^{\ell+1}, \\ \lambda_{2,D}^{\ell+1} & = & \lambda_{1,D}^{\ell} - 2\alpha\ v_1^{\ell+1}. \end{array} \right. \tag{3.16}$$

We can state and prove similar convergence results as in Proposition 3.2. We set

$$q_m^{\alpha}(R) := \frac{\sigma_2 m(R_2^{2m} + R^{2m}) - \alpha R(R_2^{2m} - R^{2m})}{\sigma_2 m(R_2^{2m} + R^{2m}) + \alpha R(R_2^{2m} - R^{2m})}. \tag{3.17}$$

Proposition 3.3. *For $f(\theta) = C_f \dfrac{\phi(\theta)}{m}$, $C_f \in \mathbb{R}$, $m \in \mathbb{N}^*$, the OSM (3.15)-(3.16) geometrically converges for all $\alpha > 0$ and the convergence rate is given by the spectral radius*

$$\hat{\rho}_m^{\alpha}(R) = \sqrt{|\ k_m^{\alpha}(R) q_m^{\alpha}(R)\ |} < 1.$$

Proof. For $f(\theta) = C_f \dfrac{\phi(\theta)}{m}$, $C_f \in \mathbb{R}$, the approximate solutions of the problem (3.15) can be written as:

$$v^{\ell+1}(r,\theta) = \left\{ \begin{array}{ll} v_1^{\ell+1}(r,\theta) = \alpha_D^{\ell+1}\ r^m \dfrac{\phi(\theta)}{m} & \text{in } \Omega_1, \\ \\ v_2^{\ell+1}(r,\theta) = \left(\beta_D^{\ell+1}\ r^m + \dfrac{\gamma_D^{\ell+1}}{r^m} \right) \dfrac{\phi(\theta)}{m} & \text{in } \Omega_2. \end{array} \right.$$

where the constants $\alpha_D^{\ell+1}$, $\beta_D^{\ell+1}$ and $\gamma_D^{\ell+1}$ are calculated by the following equations:

$$\left\{ \begin{array}{ll} v_2^{\ell+1} = f(\theta) & \text{on } \Gamma, \\ \\ \sigma_1 \dfrac{\partial v_1^{\ell+1}}{\partial \nu} + \alpha v_1^{\ell+1} = \lambda_{1,D}^{\ell} & \text{on } \Sigma, \\ \\ \sigma_2 \dfrac{\partial v_2^{\ell+1}}{\partial \nu} - \alpha v_2^{\ell+1} = \lambda_{2,D}^{\ell} & \text{on } \Sigma. \end{array} \right. \tag{3.18}$$

For $i = 1, 2$, we denote by $\hat{\lambda}_{i,D}^{\ell} = \lambda_{i,D}^{\ell} \dfrac{m}{\phi(\theta)}$. Then, from the second equation of (3.18), we obtain:

$$v_1^{\ell+1}(r,\theta) = \frac{\hat{\lambda}_{1,D}^{\ell}}{R^{m-1}(m\sigma_1 + \alpha R)} r^m \frac{\phi(\theta)}{m}, \tag{3.19}$$

and from the first and the third equation of (3.18), we get

$$v_2^{\ell+1}(r,\theta) = \left[\frac{R_2^m R^{m+1} \left((\sigma_2 m - \alpha R) C_f R^{m-1} - \hat{\lambda}_{2,D}^{\ell} R_2^m \right)}{\sigma_2 m(R_2^{2m} + R^{2m}) + \alpha R(R_2^{2m} - R^{2m})} \left(\frac{1}{r^m} - \frac{r^m}{R_2^{2m}} \right) + \frac{C_f}{R_2^m} r^m \right] \frac{\phi(\theta)}{m}.$$

From (3.16) and using the fact that $\lambda_{i,D}^{\ell+1} = \hat{\lambda}_{i,D}^{\ell+1} \dfrac{\phi(\theta)}{m}$, we obtain

$$\hat{\lambda}_{2,D}^{\ell+1} = k_m^{\alpha}(R)\ \hat{\lambda}_{1,D}^{\ell}.$$

In the same way we get

$$\hat{\lambda}_{1,D}^{\ell+1} = q_m^\alpha(R)\ \hat{\lambda}_{2,D}^{\ell} + \omega_m^\alpha(R),$$

where

$$\omega_m^\alpha(R) = \frac{4\alpha\sigma_2 m C_f R^m R_2^m}{\sigma_2 m(R_2^{2m} + R^{2m}) + \alpha R(R_2^{2m} - R^{2m})}. \tag{3.20}$$

Therefore, the DDM iterations can be written as:

$$\begin{pmatrix} \hat{\lambda}_{1,D}^{\ell+1} \\ \\ \hat{\lambda}_{2,D}^{\ell+1} \end{pmatrix} = \begin{pmatrix} 0 & q_m^\alpha(R) \\ \\ k_m^\alpha(R) & 0 \end{pmatrix} \begin{pmatrix} \hat{\lambda}_{1,D}^{\ell} \\ \\ \hat{\lambda}_{2,D}^{\ell} \end{pmatrix} + \begin{pmatrix} \omega_m^\alpha(R) \\ \\ 0 \end{pmatrix}. \tag{3.21}$$

This induction converges if and only if $\hat{\rho}_m^\alpha(R) = \sqrt{\mid k_m^\alpha(R) q_m^\alpha(R) \mid} < 1$, which is always true for $\alpha > 0$, $m > 0$ and $0 < R < R_2$. □

3.2 Combined inversion method

We now present the combined inversion algorithm that couples Algorithm 1 with OSM. Given an integer $L > 0$, an interface Σ and an initial guess $\lambda_{i,N}^0 \in L^2(\Sigma)$, $i = 1, 2$ we denote by $N^L(\Sigma, \lambda_{1,N}, \lambda_{2,N})$ the L^{th} iterate of (3.7). More precisely, we set

$$\begin{cases} N^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0) := u_1^L & \text{in } \Omega_1, \\ N^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0) := u_2^L & \text{in } \Omega_2, \end{cases}$$

where (u_1^ℓ) and (u_2^ℓ), $\ell = 1, \ldots, L$ verify the induction (3.7). We also set

$$\Lambda_{i,N}^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0) := \lambda_{i,N}^L \text{ on } \Sigma,\ i = 1, 2,$$

where $\lambda_{i,N}^L$ is the iterate number L of (3.7).

Similarly, for some initial guess $\lambda_{i,D}^0 \in L^2(\Sigma)$, $i = 1, 2$ we define:

$$\begin{cases} D^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0) := v_1^L & \text{in } \Omega_1, \\ D^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0) := v_2^L & \text{in } \Omega_2, \end{cases}$$

where (v_1^ℓ) and (v_2^ℓ), $\ell = 1, \ldots, L$ verify the induction (3.15) and by

$$\Lambda_{i,D}^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0) := \lambda_{i,D}^L \text{ on } \Sigma,\ i = 1, 2,$$

where $\lambda_{i,D}^L$ is the iterate number L of (3.15).

Roughly speaking, the combined algorithm consists in replacing $u_{\sigma(R^k)}$ and $v_{\sigma(R^k)}$ respectively with $N^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0)$ and $D^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0)$ where $\Sigma = \mathcal{T}_n(R^k)$ as in Algorithm 1. The main ambiguity in the construction of the induction is the update for $\lambda_{i,N}$ and $\lambda_{i,D}$ on $\Sigma = \mathcal{T}_n(R^{k+1})$ using the boundary values of the solutions at previous iterate. We choose to transport these values on the new update of the interface (i.e $\Sigma = \mathcal{T}_n(R^{k+1})$) using the same mapping that transforms $\mathcal{T}_n(R^k)$ into $\mathcal{T}_n(R^{k+1})$. We explicit this for starlike interfaces as in Algorithm 1 where we suppose that the number of subdivisions n of the interface is kept fixed during iterations. A function λ defined on $\Sigma = \Sigma_R$ can be parameterized on S_j, $j = 0, \ldots, n-1$ as a function of the variable $t \in [0, 1]$ as follows

$$\lambda(t) \equiv \lambda(\hat{M}_j(t)) \tag{3.22}$$

where $\hat{M}_j(t)$ is given by (2.2). The gradient flow as defined by Algorithm 1 preserves the parametrization of Σ in terms of S_j. Therefore the update for the boundary values on Σ can be written easily in terms of the variable $t \in [0,1]$ since the latter is independent from the iteration index.

There is also an ambiguity in defining the partial shape derivative given by (2.5) since the OSM iterates are not continuous across Σ_R nor are the fluxes in general. One can either evaluate the shape gradient with respect to $N^L(\Sigma, \lambda^0_{1,N}, \lambda^0_{2,N})$ and $D^L(\Sigma, \lambda^0_{1,D}, \lambda^0_{2,D})$ or approximate the gradient of the exact cost functional using these approximate solutions of the direct problems. The first option would require the introduction of two adjoint problems and therefore may render the method more costly. This is why we adopt the second approach, that indeed lead to an incorrect gradient at first iterations, but this gradient becomes close to the exact one as the iterations number increases. More precisely, the gradient expression given by (2.5) requires the trace and the normal trace values of the solutions to the Dirichlet and Neumann problems which are continuous for the exact solutions. However, these values are no longer the same if we use the restriction of solutions to Ω_1 or Ω_2 coming from incomplete OSM iterations. In the following algorithm, we choose to apply (2.5) using the traces and the normal traces of $u_{\sigma(R)_{|\Omega_1}}$ and $v_{\sigma(R)_{|\Omega_1}}$.

Algorithm 2: A combined inversion algorithm

- Fix the number of parameters $n \in \mathbb{N}^*$ that serve to define the starlike interface.
- Consider an initial guess $R^0 \in (\mathbb{R}_+^*)^n$, the initial interface $\Sigma = \Sigma_{R^0} = \mathcal{T}_n(R^0)$ and the corresponding conductivity $\sigma(R^0) = \sigma_1 \chi_{\Omega_{1,R^0}} + \sigma_2 \chi_{\Omega_{2,R^0}}$ as defined above.
- Choose as initial boundary values:

$$\lambda_{i,N}(\Sigma) = 0,\ \lambda_{i,D}(\Sigma) = 0 \text{ on } \Sigma,\ i = 1, 2.$$

- k=0;

repeat *until* $k \leq$ *maximum number of iterations*

1. Set
$$\lambda_{i,N}^0 = \lambda_{i,N}(\Sigma),\ \lambda_{i,D}^0 = \lambda_{i,D}(\Sigma) \text{ on } \Sigma,\ i = 1, 2.$$
2. Use L iterations of OSM to evaluate
$$u_{\sigma(R^k)} = N^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0),\ \lambda_{i,N}^L(R^k) = \Lambda_{i,N}^L(\Sigma, \lambda_{1,N}^0, \lambda_{2,N}^0).$$
$$v_{\sigma(R^k)} = D^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0),\ \lambda_{i,D}^L(R^k) = \Lambda_{i,D}^L(\Sigma, \lambda_{1,D}^0, \lambda_{2,D}^0).$$
3. Evaluate $\dfrac{\partial \mathcal{J}}{\partial R_j}(R^k)$, for $j = 0, \ldots, n-1$ using formula (2.5) where the boundary values are calculated using
$$u_{\sigma(R^k)_{|\Omega_1}} \text{ and } v_{\sigma(R^k)_{|\Omega_1}}.$$
4. Update $\Sigma = \mathcal{T}_n(R^{k+1})$ with
$$R_j^{k+1} := R_j^k - \tau \frac{\partial \mathcal{J}}{\partial R_j}(R^k),\ j = 0, \ldots, n-1,$$
where $\tau > 0$ is chosen sufficiently small. (A step adaptation can be incorporated here but only after a few iterations, when the gradient becomes sufficiently accurate and provides a descent direction).
5. Update the interface values on S_j as
$$\lambda_{i,N}(\Sigma)(\hat{M}_j(t)) = \lambda_{i,N}^L(R^k)(t),\ t \in [0, 1],$$
$$\lambda_{i,D}(\Sigma)(\hat{M}_j(t)) = \lambda_{i,D}^L(R^k)(t),\ t \in [0, 1],$$
following the identification (3.22).
6. $R^k = R^{k+1}$.
7. $k = k + 1$.

end

As indicated earlier, in step 3 of Algorithm 2 we could have made the choice to compute the derivative using $u_{\sigma(R^k)_{|\Omega_2}}$ and $v_{\sigma(R^k)_{|\Omega_2}}$. This would not affect the theoretical or numerical results below. We keep the choice made in Algorithm 2 in the reminder of this paper for the sake of conciseness.

3.3 Local convergence analysis of Algorithm 2 for circular domains

This section is dedicated to the convergence analysis of Algorithm 2 in the case of circular domains as in Section (2.3) and Section (3.1.2) and for $L = 1$, i.e we perform only one DDM step per iteration. With the notation of Section 2.3 and Algorithm 2, the iterative scheme leads to the following. The boundary values at iteration k are of the form

$$\lambda_{i,N}^L(R^k) = \hat{\lambda}_{i,N}^L(R^k)\frac{\phi(\theta)}{m}, \quad \lambda_{i,D}^L(R^k) = \hat{\lambda}_{i,D}^L(R^k)\frac{\phi(\theta)}{m},$$

for some constants $\hat{\lambda}_{i,N}^L(R^k)$ and $\hat{\lambda}_{i,D}^L(R^k)$ and for $i = 1, 2$. Therefore according to equations (3.11) and (3.19), the solutions after one step DDM iterate are given in Ω_1 by

$$u_{\sigma(R^k)}(r,\theta) = \alpha_N^L(R^k)r^m\frac{\phi(\theta)}{m}, \quad v_{\sigma(R^k)}(r,\theta) = \alpha_D^L(R^k)r^m\frac{\phi(\theta)}{m},$$

for $R^k < r < R_2$ with

$$\begin{cases} \alpha_N^L(R) := \dfrac{\hat{\lambda}_{1,N}^L(R)}{R^{m-1}(m\sigma_1 + \alpha R)}, \\ \alpha_D^L(R) := \dfrac{\hat{\lambda}_{1,D}^L(R)}{R^{m-1}(m\sigma_1 + \alpha R)}. \end{cases}$$

Consequently, by step 3 of Algorithm 2, the gradient is approximated by

$$\frac{\partial \mathcal{J}}{\partial R}(R^k) \simeq \frac{\pi m^2(\sigma_1^2 - \sigma_2^2)R^k}{\sigma_2(\sigma_1 m + \alpha R^k)^2}(|\hat{\lambda}_{1,D}^L(R^k)|^2 - |\hat{\lambda}_{1,N}^L(R^k)|^2).$$

The iterative scheme of Algorithm 2 now reads

$$R^{k+1} = R^k - \tau\frac{\pi m^2(\sigma_1^2 - \sigma_2^2)R^k}{\sigma_2(\sigma_1 m + \alpha R^k)^2}(|\hat{\lambda}_{1,D}^L(R^k)|^2 - |\hat{\lambda}_{1,N}^L(R^k)|^2) \tag{3.23}$$

and the updates for the boundary values are given by

$$\hat{\lambda}_{1,N}^L(R^{k+1}) = p_m^\alpha(R^k)\ \hat{\lambda}_{2,N}^L(R^k) + \eta_m^\alpha(R^k), \tag{3.24}$$
$$\hat{\lambda}_{2,N}^L(R^{k+1}) = k_m^\alpha(R^k)\ \hat{\lambda}_{1,N}^L(R^k), \tag{3.25}$$
$$\hat{\lambda}_{1,D}^L(R^{k+1}) = q_m^\alpha(R^k)\ \hat{\lambda}_{2,D}^L(R^k) + \omega_m^\alpha(R^k), \tag{3.26}$$
$$\hat{\lambda}_{2,D}^L(R^{k+1}) = k_m^\alpha(R^k)\ \hat{\lambda}_{1,D}^L(R^k), \tag{3.27}$$

where $p_m^\alpha(R^k)$, $\eta_m^\alpha(R^k)$, $k_m^\alpha(R^k)$, $q_m^\alpha(R^k)$ and $\omega_m^\alpha(R^k)$ are respectively given by (3.9), (3.12), (3.8), (3.17) and (3.20).

If we set $X^k := (R^k, \hat{\lambda}_{1,N}^L(R^k), \hat{\lambda}_{2,N}^L(R^k), \hat{\lambda}_{1,D}^L(R^k), \hat{\lambda}_{2,D}^L(R^k))$, then the iterative scheme formed by (3.23)-(3.27) can be synthetically written as

$$X^{k+1} = G(X^k),$$

where $G :]0, R_2[\times\mathbb{R}^4 \longrightarrow \mathbb{R}^5$, $X = (x, y, z, t, h) \mapsto (g_1(X), g_2(X), g_2(X), g_2(X), g_2(X))$ is given by

$$\begin{aligned} g_1(X) &:= x - \tau\frac{\pi m^2(\sigma_1^2 - \sigma_2^2)x}{\sigma_2(\sigma_1 m + \alpha x)^2}(t^2 - y^2), \\ g_2(X) &:= p_m^\alpha(x)z + \eta_m^\alpha(x), \\ g_3(X) &:= k_m^\alpha(x)y, \\ g_4(X) &:= q_m^\alpha(x)h + \omega_m^\alpha(x), \\ g_5(X) &:= k_m^\alpha(x)t. \end{aligned}$$

From the analysis of section (3.1.2) we observe that the sequence $\hat{\lambda}_{i,N}^{\ell}(\overline{R})$, $i = 1, 2$ converges to $\hat{\lambda}_{i,N}^{\infty}$, $i = 1, 2$, where

$$\hat{\lambda}_{i,N}^{\infty} = \frac{2R_2^{m+1}\overline{R}^{m-1}}{\sigma_1\left(R_2^{2m} + \overline{R}^{2m}\right) + \sigma_2\left(R_2^{2m} - \overline{R}^{2m}\right)}\psi_i$$

with

$$\psi_1 = (\sigma_1 m + \alpha\overline{R}) \quad \text{and} \quad \psi_2 = (\sigma_1 m - \alpha\overline{R}).$$

Similarly, the sequence $\hat{\lambda}_{i,D}^{\ell}(\overline{R})$, $i = 1, 2$ converges to $\hat{\lambda}_{i,D}^{\infty}$, $i = 1, 2$, where

$$\hat{\lambda}_{i,D}^{\infty} = \frac{2\sigma_2 C_f R_2^m \overline{R}^{m-1}}{\sigma_1\left(R_2^{2m} - \overline{R}^{2m}\right) + \sigma_2\left(R_2^{2m} + \overline{R}^{2m}\right)}\psi_i.$$

Moreover, we have

$$\begin{cases} \hat{\lambda}_{1,N}^{\infty} = p_m^\alpha(\overline{R})\hat{\lambda}_{2,N}^{\infty} + \eta_m^\alpha(\overline{R}) \\ \hat{\lambda}_{2,N}^{\infty} = k_m^\alpha(\overline{R})\hat{\lambda}_{1,N}^{\infty} \\ \hat{\lambda}_{1,D}^{\infty} = q_m^\alpha(\overline{R})\hat{\lambda}_{2,D}^{\infty} + w_m^\alpha(\overline{R}) \\ \hat{\lambda}_{2,D}^{\infty} = k_m^\alpha(\overline{R})\hat{\lambda}_{1,D}^{\infty} \end{cases}$$

Using the expression of C_f given by (2.7) one can easily check that $\hat{\lambda}_{i,N}^{\infty} = \hat{\lambda}_{i,D}^{\infty}$ and then $\overline{X} = (\overline{R}, \hat{\lambda}_{1,N}^{\infty}, \hat{\lambda}_{2,N}^{\infty}, \hat{\lambda}_{1,D}^{\infty}, \hat{\lambda}_{2,D}^{\infty}) = (\overline{R}, \hat{\lambda}_{1,N}^{\infty}, \hat{\lambda}_{2,N}^{\infty}, \hat{\lambda}_{1,N}^{\infty}, \hat{\lambda}_{2,N}^{\infty})$ is a fixed point of G. We first establish in the following lemma that $\overline{X}$ is the unique fixed point of G.

Lemma 3.4. *The point $\overline{X}$ defined above is the unique fixed point of G .*

Proof. Let $X = (x, y, z, t, h)$, $x > 0$, be a fixed point of G. Then we have from $g_1(X) = x$ that $t = y$ and the other equalities imply

$$\begin{cases} y = p_m^\alpha(x)z + \eta_m^\alpha(x), \\ z = k_m^\alpha(x)y, \\ t = q_m^\alpha(x)h + w_m^\alpha(x) \\ h = k_m^\alpha(x)t. \end{cases}$$

Consequently

$$\begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} 0 & p_m^\alpha(x) \\ k_m^\alpha(x) & 0 \end{pmatrix}\begin{pmatrix} y \\ z \end{pmatrix} + \begin{pmatrix} \eta_m^\alpha(x) \\ 0 \end{pmatrix}$$

and

$$\begin{pmatrix} t \\ \\ h \end{pmatrix} = \begin{pmatrix} 0 & q_m^\alpha(x) \\ \\ k_m^\alpha(x) & 0 \end{pmatrix} \begin{pmatrix} t \\ \\ h \end{pmatrix} + \begin{pmatrix} w_m^\alpha(x) \\ \\ 0 \end{pmatrix}.$$

Then we get

$$\begin{cases} y = \dfrac{2R_2^{m+1}x^{m-1}(\sigma_1 m + \alpha x)}{\sigma_1\left(R_2^{2m} + x^{2m}\right) + \sigma_2\left(R_2^{2m} - x^{2m}\right)} \\ t = \dfrac{2\sigma_2 C_f R_2^m x^{m-1}(\sigma_1 m + \alpha x)}{\sigma_1\left(R_2^{2m} - x^{2m}\right) + \sigma_2\left(R_2^{2m} + x^{2m}\right)}. \end{cases}$$

The equality $y = t$ then implies

$$C_f = \frac{R_2}{\sigma_2}\frac{\left[\sigma_1(R_2^{2m} - x^{2m}) + \sigma_2(R_2^{2m} + x^{2m})\right]}{\left[\sigma_1(R_2^{2m} + x^{2m}) + \sigma_2(R_2^{2m} - x^{2m})\right]}.$$

Since the function g defined on $]0, R_2[$ by

$$g(x) = \frac{R_2}{\sigma_2}\frac{\left[\sigma_1(R_2^{2m} - x^{2m}) + \sigma_2(R_2^{2m} + x^{2m})\right]}{\left[\sigma_1(R_2^{2m} + x^{2m}) + \sigma_2(R_2^{2m} - x^{2m})\right]}$$

is one to one, we deduce from the expression of C_f given by (2.7) that $x = \overline{R}$. This finally gives $X = (\overline{R}, \hat{\lambda}_{1,N}^\infty, \hat{\lambda}_{2,N}^\infty, \hat{\lambda}_{1,D}^\infty, \hat{\lambda}_{2,D}^\infty) = \overline{X}$. □

Theorem 3.1. *There exists $\alpha_0 > 0$ such that for all $\alpha \in]0, \alpha_0[$ Algorithm 2 with $L = 1$ is locally convergent for all $\tau \in]0, \delta_\alpha[$ for some $\delta_\alpha > 0$.*

Proof. For short notation, let us set $\overline{x} = \overline{R}$, $\overline{y} = \hat{\lambda}_{1,N}^\infty$, and

$$k := k_m^\alpha(\overline{x}) = \frac{\sigma_1 m - \alpha\overline{x}}{\sigma_1 m + \alpha\overline{x}} = \frac{\psi_2}{\psi_1}.$$

Then from the previous Lemma, $\overline{X} = (\overline{x}, \overline{y}, k\overline{y}, \overline{y}, k\overline{y})$ is the unique fixed point of G. In order to prove the local convergence we shall establish that G is a contraction in a neighborhood of $\overline{X}$. This requires the study of the Jacobian matrix $DG(\overline{X})$. The latter is given by

$$DG(\overline{X}) = \begin{pmatrix} 1 & d\tau & 0 & -d\tau & 0 \\ a & 0 & p & 0 & 0 \\ b & k & 0 & 0 & 0 \\ c & 0 & 0 & 0 & q \\ b & 0 & 0 & k & 0 \end{pmatrix}$$

where we have set for short notation

$$
\begin{aligned}
d &:= \frac{1}{\tau}\frac{\partial g_1}{\partial y}(\overline{X}) = \frac{2\pi m^2(\sigma_1^2-\sigma_2^2)\overline{x}}{\sigma_2(\sigma_1 m+\alpha\overline{x})^2}\overline{y} \\
a &:= \frac{\partial g_2}{\partial x}(\overline{X}) = \frac{-2m\alpha\sigma_2(4mR_2^{2m}\overline{x}^{2m}-\overline{x}^{4m}+R_2^{4m})}{\left[\sigma_2 m(R_2^{2m}-\overline{x}^{2m})+\alpha\overline{x}(R_2^{2m}+\overline{x}^{2m})\right]^2}k\overline{y} \\
&+ \frac{4m\alpha R_2^{m+1}\overline{x}^{m-1}\left[(\sigma_2 m^2-\alpha\overline{x})(R_2^{2m}+\overline{x}^{2m})+\alpha\overline{x}m(R_2^{2m}-\overline{x}^{2m})\right]}{\left[\sigma_2 m(R_2^{2m}-\overline{x}^{2m})+\alpha\overline{x}(R_2^{2m}+\overline{x}^{2m})\right]^2} \\
p &:= \frac{\partial g_2}{\partial z}(\overline{X}) = p_m^\alpha(\overline{x}) \\
b &:= \frac{\partial g_3}{\partial x}(\overline{X}) = \frac{-2\sigma_1 m\alpha}{(\sigma_1 m+\alpha\overline{x})^2}\overline{y} \\
c &:= \frac{\partial g_4}{\partial x}(\overline{X}) = \frac{2m\alpha\sigma_2(4mR_2^{2m}\overline{x}^{2m}+\overline{x}^{4m}-R_2^{4m})}{\left[\sigma_2 m(R_2^{2m}+\overline{x}^{2m})+\alpha\overline{x}(R_2^{2m}-\overline{x}^{2m})\right]^2}k\overline{y} \\
&+ \frac{4m\sigma_2\alpha C_f R_2^m\overline{x}^{m-1}\left[(\sigma_2 m^2-\alpha\overline{x})(R_2^{2m}-\overline{x}^{2m})+\alpha\overline{x}m(R_2^{2m}+\overline{x}^{2m})\right]}{\left[\sigma_2 m(R_2^{2m}+\overline{x}^{2m})+\alpha\overline{x}(R_2^{2m}-\overline{x}^{2m})\right]^2}, \\
q &:= \frac{\partial g_4}{\partial h}(\overline{X}) = q_m^\alpha(\overline{x}).
\end{aligned}
$$

For every $(\lambda,\tau)\in\mathbb{R}\times\mathbb{R}$, we define the characteristic polynomial of $DG(\overline{X})$

$$
P(\lambda,\tau) := det(DG(\overline{X})-\lambda I_5) = \begin{vmatrix} 1-\lambda & d\tau & 0 & -d\tau & 0 \\ a & -\lambda & p & 0 & 0 \\ b & k & -\lambda & 0 & 0 \\ c & 0 & 0 & -\lambda & q \\ b & 0 & 0 & k & -\lambda \end{vmatrix}
$$

and we denote by $\lambda_i(\tau)\in\mathbb{C}$, $i=1,\ldots,5$, the eigenvalues of $DG(\overline{X})$. Then, we have

$$
P(\lambda,\tau) = (1-\lambda)(\lambda^2-kp)(\lambda^2-kq)-\tau\, d\left[(c-a)\lambda^3+b(q-p)\lambda^2+(aq-cp)k\lambda\right],
$$

and $P(\lambda_i(\tau),\tau)=0$ for $i=1,\ldots,5$. Let $\overline{\rho}(\tau):=\max\{|\lambda_i(\tau)|;\quad i=1,\ldots,5\}$ be the spectral radius of $DG(\overline{X})$. We remark that for $\tau=0$, we have

$$
P(\lambda,0)=(1-\lambda)(\lambda^2-kp)(\lambda^2-kq),
$$

and therefore, the eigenvalues of the $DG(\overline{X})$ for $\tau=0$ are given by

$$
(\lambda_1(0),\lambda_2(0),\lambda_3(0),\lambda_4(0),\lambda_5(0)) = \left(1,\sqrt{kp},-\sqrt{kp},\sqrt{kq},-\sqrt{kq}\right)
$$

where $|kp|<1$, $|kq|<1$, and $kq\neq kp$ if α sufficiently small. Then, for $\tau=0$ and α sufficiently small, all eigenvalues of $DG(\overline{X})$ are simple and verify: $|\lambda_1(0)|=1$, and $|\lambda_i(0)|<1$ for $i\in\{2,3,4,5\}$. Consequently, there exists a small enough number $\delta>0$ such that all eigenvalues of $DG(\overline{X})$ are simple for all $\tau\in]0,\delta[$ and

$$
\lim_{\tau\to 0}\lambda_i(\tau)=\lambda_i(0)\ \ \forall\, i\in\{1,2,3,4,5\}.
$$

Thus one can choose $\delta > 0$ sufficiently small such that $|\lambda_i(\tau)| < 1$ for $i \in \{2,3,4,5\}$, $\overline{\rho}(\tau) = |\lambda_0(\tau)|$ for all $\tau \in]0,\delta[$. Moreover $\lim_{\tau \longrightarrow 0} \overline{\rho}(\tau) = 1$. To show that if $\overline{\rho}(\tau) < 1$ for small enough $\tau > 0$, we use a first order Taylor expansion of $(\lambda,\tau) \mapsto P(\lambda,\tau)$ in a neighborhood of the point $(1,0)$. Indeed

$$\begin{aligned} P(\lambda,\tau) &= P(1,0) + \frac{\partial P}{\partial \lambda}(1,0)\,(\lambda-1) + \frac{\partial P}{\partial \tau}(1,0)\,\tau + \epsilon(\lambda,\tau) \\ &= C_1(1-\lambda) - C_2\tau + \epsilon(\lambda,\tau) \end{aligned}$$

with $\lim_{(\lambda,\tau)\to(1,0)} \dfrac{|\epsilon(\lambda,\tau)|}{\sqrt{(1-\lambda)^2+\tau^2}} = 0$, and

$$\left\{ \begin{array}{l} C_1 := (1-kp)(1-kq), \\ C_2 := d\left[(c-a)(1-kp) + (q-p)(b+ak)\right]. \end{array} \right. \tag{3.28}$$

This shows that for τ sufficiently small

$$\lambda_0(\tau) = 1 - \frac{C_2}{C_1}\tau + o(\tau). \tag{3.29}$$

Clearly $C_1 > 0$ from the conditions $|kp| < 1$ and $|kq| < 1$. Let $\alpha \in]0,\alpha_0[$ where α_0 is as in Lemma 3.5. Then we also have that $C_2 > 0$. We then infer from equation (3.29) the existence of $\delta_\alpha > 0$ such that $\overline{\rho}(\tau) = |\lambda_0(\tau)| < 1$ for all $\tau \in]0,\delta_\alpha[$. This proves that G is a contraction for $\tau \in]0,\delta_\alpha[$, which gives the desired result. □

Lemma 3.5. *There exists $\alpha_0 > 0$ such that $C_2 > 0$ for all $0 < \alpha < \alpha_0$ where $C_2 = C_2(\alpha)$ is given by* (3.28).

Proof. We first observe that

$$\overline{y} \;\sim\; k\overline{y} \sim \frac{2m\sigma_1 R_2^{m+1}\overline{R}^{m-1}}{\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)} \text{ as } \alpha \to 0.$$

Then, using a first order Taylor expansion of $a,\ b,\ c,\ d,\ k,\ p,\ q$, in a neighborhood of $\alpha = 0$ we get the following equivalences as $\alpha \to 0$

$$d \;\sim\; \frac{2\pi\left(\sigma_1^2-\sigma_2^2\right)\overline{x}}{\sigma_1^2\sigma_2}\overline{y} \sim \frac{4m\pi(\sigma_1^2-\sigma_2^2)R_2^{m+1}\overline{R}^{m}}{\sigma_1\sigma_2\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]} \tag{3.30}$$

$$
\begin{aligned}
b &\sim \frac{-2\alpha}{m\sigma_1}\overline{y} \sim \frac{-4\alpha R_2^{m+1}\overline{R}^{m-1}}{\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)} \\
k &\sim 1-\frac{2\alpha\overline{R}}{m\sigma_1} \\
p &\sim 1-\frac{2\alpha\overline{R}\left(R_2^{2m}+\overline{R}^{2m}\right)}{m\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)} \\
q &\sim 1-\frac{2\alpha\overline{R}\left(R_2^{2m}-\overline{R}^{2m}\right)}{m\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)} \\
a &\sim \frac{-2\alpha\left(4mR_2^{2m}\overline{R}^{2m}-\overline{R}^{4m}+\overline{R_2}^{4m}\right)}{m\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)^2}\overline{y}+\frac{4m\alpha R_2^{m+1}\overline{R}^{m-1}\left(R_2^{2m}+\overline{R}^{2m}\right)}{\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)^2} \\
&\sim \frac{4\alpha R_2^{m+1}\overline{R}^{m-1}\left[m\sigma_1\left(R_2^{2m}-\overline{R}^{2m}\right)+(m\sigma_2-\sigma_1)\left(R_2^{2m}+\overline{R}^{2m}\right)\right]}{\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]} \\
c &\sim \frac{2\alpha\left(4mR_2^{2m}\overline{R}^{2m}+\overline{R}^{4m}-\overline{R_2}^{4m}\right)}{m\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)^2}\overline{y}+\frac{4m\alpha C_f R_2^m\overline{R}^{m-1}\left(R_2^{2m}-\overline{R}^{2m}\right)}{\left(R_2^{2m}+\overline{R}^{2m}\right)^2} \\
&\sim \frac{4\alpha R_2^{m+1}\overline{R}^{m-1}\left[m\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+(m\sigma_2-\sigma_1)\left(R_2^{2m}-\overline{R}^{2m}\right)\right]}{\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]}.
\end{aligned}
$$

Then we infer that

$$
\begin{aligned}
c-a &\sim \frac{16\alpha(\sigma_1-m\sigma_2)R_2^{3m+1}\overline{R}^{3m-1}}{\sigma_2\left(R_2^{4m}-\overline{R}^{4m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]} \\
1-kp &\sim \frac{2\alpha\overline{R}\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]}{m\sigma_1\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)} \qquad (3.31) \\
1-kq &\sim \frac{2\alpha\overline{R}\left[\sigma_1\left(R_2^{2m}-\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)\right]}{m\sigma_1\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)} \qquad (3.32)
\end{aligned}
$$

$$
\begin{aligned}
q-p &\sim \frac{8\alpha R_2^{2m}\overline{R}^{2m+1}}{m\sigma_2\left(R_2^{4m}-\overline{R}^{4m}\right)} \\
b+ak\sim b+a &\sim \frac{4\alpha R_2^{m+1}\overline{R}^{m-1}\left[(m\sigma_1-\sigma_2)\left(R_2^{2m}-\overline{R}^{2m}\right)+(m\sigma_2-\sigma_1)\left(R_2^{2m}+\overline{R}^{2m}\right)\right]}{\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]}.
\end{aligned}
$$

Consequently,

$$(c-a)(1-kp) \sim \frac{32\alpha^2(\sigma_1-m\sigma_2)R_2^{3m+1}\overline{R}^{3m}}{m\sigma_1\sigma_2^2\left(R_2^{4m}-\overline{R}^{4m}\right)\left(R_2^{2m}-\overline{R}^{2m}\right)}$$

$$(q-p)(b+ak) \sim \frac{32\alpha^2R_2^{3m+1}\overline{R}^{3m}\left[(m\sigma_1-\sigma_2)\left(R_2^{2m}-\overline{R}^{2m}\right)+(m\sigma_2-\sigma_1)\left(R_2^{2m}+\overline{R}^{2m}\right)\right]}{m\sigma_2^2\left(R_2^{4m}-\overline{R}^{4m}\right)\left(R_2^{2m}-\overline{R}^{2m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]},$$

and from equations (3.28) and (3.30), we obtain

$$C_2 \sim \frac{128\pi\alpha^2m\left(\sigma_1^2-\sigma_2^2\right)^2R_2^{4m+2}\overline{R}^{4m}}{\sigma_1^2\sigma_2^3\left(R_2^{4m}-\overline{R}^{4m}\right)\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]^2} > 0.$$

□

Remark 3.2. *Using equations (3.28), (3.31) and (3.32), we obtain*

$$C_1 \sim \frac{4\alpha^2\overline{R}^2\left[\sigma_1\left(R_2^{2m}+\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}-\overline{R}^{2m}\right)\right]\left[\sigma_1\left(R_2^{2m}-\overline{R}^{2m}\right)+\sigma_2\left(R_2^{2m}+\overline{R}^{2m}\right)\right]}{m^2\sigma_1^2\sigma_2^2\left(R_2^{4m}-\overline{R}^{4m}\right)}$$

as $\alpha \to 0$. *Therefore,*

$$\frac{C_2}{C_1} \sim \mathcal{J}''(\overline{R}) \text{ as } \alpha \to 0. \tag{3.33}$$

This shows that the local convergence of Algorithm 2 is equivalent to the local convergence of Algorithm 1 in this specific case. In fact, some calculations with Maple for $m = 1$ *and* $m = 2$ *show that we have exact equality* $C_2/C_1 = \mathcal{J}''(\overline{R})$ *for all* $\alpha > 0$ *and* $\overline{R} > 0$. *We conjecture that this also holds true for all* $m > 0$.

4 Numerical experiments and validation

The goal of this section is to test the efficiency of Algorithm 2 in comparison with the standard descent gradient described in Algorithm 1. We shall employ synthetic data numerically generated using a finite elements solver designed with the help of FreeFem++ [24]. Indeed, we use a direct solver to generate the data, while the OSM is used in the inversion for Algorithm 2. We further avoid any inverse crime by making sure that the meshes used for generating the data have no connections with the ones used in the inversion. Actually, the latter vary during iterations since the interface Σ changes. Moreover, in most of the examples below the geometry of Σ can not be exactly represented by the parametrization (2.3) used in the inversion algorithms.

For all the examples below, the domain Ω is the open disk of center $(0,0)$ and radius $R_2 = 2$, the current flux $\phi(\theta) = \cos(\theta)$, $\theta \in [0, 2\pi]$, the conductivity parameters $\sigma_1 = 1$, $\sigma_2 = 2$ and the OSM parameter $\alpha = 1$. The measured data f is represented by the values f_i, $i = 1, \ldots, N_\Gamma$ of the numerical solution u_{num} at the nodes belonging to Γ. In order to simulate noise in the data f we artificially corrupt the computed values f_i with random noise as follows

$$f_i^\epsilon := f_i + \epsilon(1-2r_i)f_i,\ i = 1, \ldots, N_\Gamma,$$

where r_i are randomly chosen between 0 and 1 and ϵ denotes the noise level. In addition to representing the obtained geometrical reconstructions we shall also give the evolution of the dimensionless square-root cost functional $\sqrt{\mathcal{J}/\mathcal{J}_0}$ during iterations where $\mathcal{J}$ is given by (2.4) and

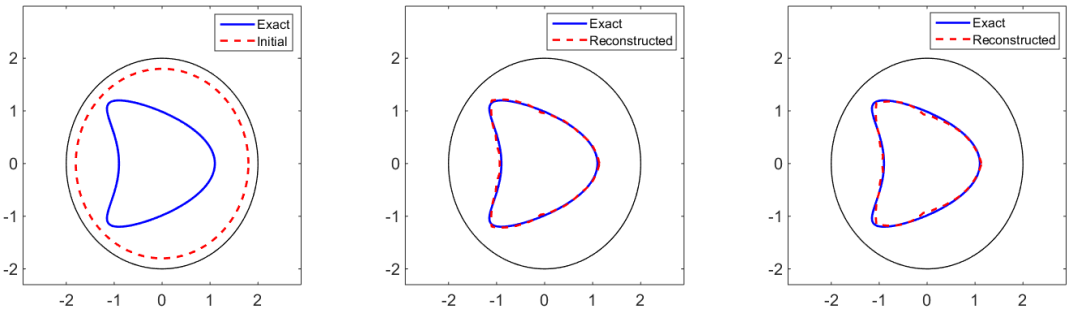
$$\mathcal{J}_0 := \int_\Omega \sigma |\nabla u_{\text{num}}|^2 dx.$$

4.1 Discussion of Algorithm 2 for a kite shape

In the first example we choose Σ to be a kite defined by

$$x(t) = \cos(t) + 0.5\cos(2t) - 0.4 \text{ and } y(t) = 1.2\sin(t),\ t \in [0, 2\pi]. \tag{4.1}$$

For the inversion algorithms we use the parametrization (2.3) with $n = 19$. The initial guess is $R_j^0 = 1.8$, for $j = 0, \ldots, n-1$. The results of the inversion are given in Figure 3(b) for Algorithm 1 and Figure 3(c) for Algorithm 2 with $L = 1$, i.e only one OSM step is used at each gradient descent iteration. We observe that we qualitatively obtain the same accuracy for both algorithms. The evolution of the cost functional is depicted in Figure 4(left). We remark that the cost functional increases in the first iterations for Algorithm 2, which means that the approximated OSM solution is not yet sufficiently close to the exact one and therefore the approximate gradient is not yet a descent direction. This is corrected as the iteration number increases. Whence the iteration number is sufficiently large we notice that the speed of convergence represented by the slope of the curves is roughly the same between the two algorithms. This indeed shows the superiority and relevance of Algorithm 2 which achieves comparative performances with potentially much cheaper numerical cost per iteration. Figure 4(middle) and (right) show the effect of increasing the number L of OSM steps. We naturally observe that as this number increases, Algorithm 2 becomes closer to Algorithm 1. Let us also mention that, other numerical tests not reported here suggest that this observation also holds when we fix L, for instance $L = 1$, and decrease the descent step τ (see also Figure 9(right)).



(a) Initial guess (b) Algorithm 1 (c) Algorithm 2

Figure 3: Comparison between Algorithm 1 and Algorithm 2 with $L = 1$ for the case of the kite parameterized by (4.1) and for noise free data. The exact shape and initial guess are shown in Figure (a). The gradient descent parameter $\tau = 0.05$ for both algorithms.

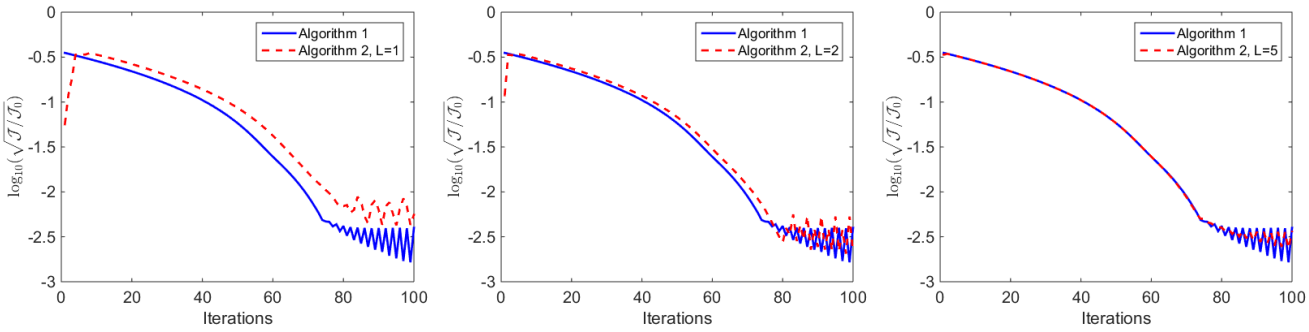


Figure 4: Comparison of the evolution of $\log_{10}(\sqrt{\mathcal{J}/\mathcal{J}_0})$ between Algorithm 1 and Algorithm 2 with $L = 1$ (left), $L = 2$ (middle) and $L = 5$ (right) for the example shown in Figure 3. The gradient descent parameter is $\tau = 0.05$.

Figure 5 shows the reconstructions obtained by Algorithm 2 for the example discussed in Figure 3 but for noisy data with noise level $\epsilon = 1\%$ (left), $\epsilon = 3\%$ (middle) and $\epsilon = 5\%$ (right). We observe robustness of the obtained results and a good accuracy which is very similar to the one obtains with Algorithm 1 (not reported here).

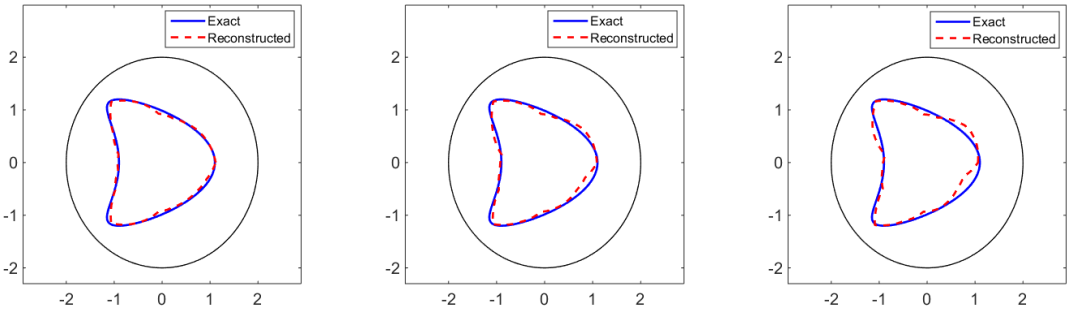


Figure 5: Reconstructions obtained by Algorithm 2 with $L = 1$ for the example discussed in Figure 3 but for noisy data with noise level $\epsilon = 1\%$ (left), $\epsilon = 3\%$ (middle) and $\epsilon = 5\%$ (right).

4.2 The example of other geometries

We report in Figure 6 and Figure 7 the reconstructions obtained by Algorithm 2, $L = 1$ for other geometries, keeping the same parameters as in Example 4.1 and with noise levels $\epsilon = 3\%$ and $\epsilon = 5\%$. The parametrizations of these geometries are given by Table 1.

Geometry type	Parametrization $(x(t), y(t))$
Peanut shaped	$1.5\sqrt{\cos^2(t) + 0.23\sin^2(t)}(\cos(t), \sin(t))$
Pear	$\frac{5.5 + \sin(3t)}{5}(\cos(t), \sin(t))$
Star	$(1.2 + 0.4\sin(5t))(\cos(t), \sin(t))$

Table 1: The boundary parametrization of the geometries.

We clearly observe good performances in terms of robustness and accuracy.

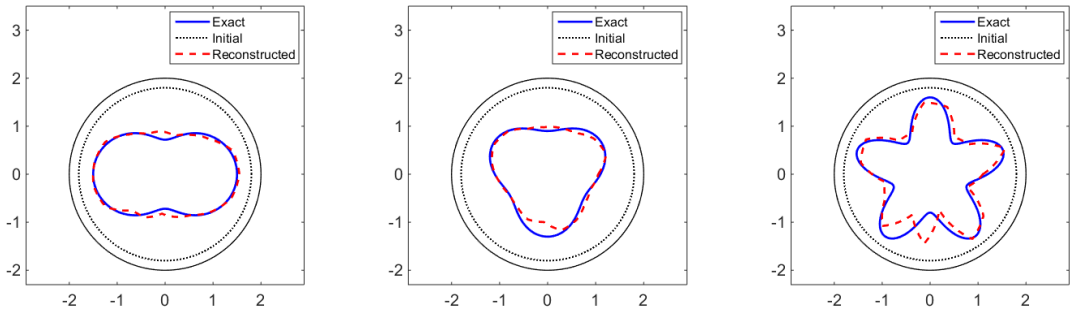


Figure 6: Reconstructions obtained by Algorithm 2 for the geometries discussed in Table 1 but for noisy data with noise level $\epsilon = 3\%$.

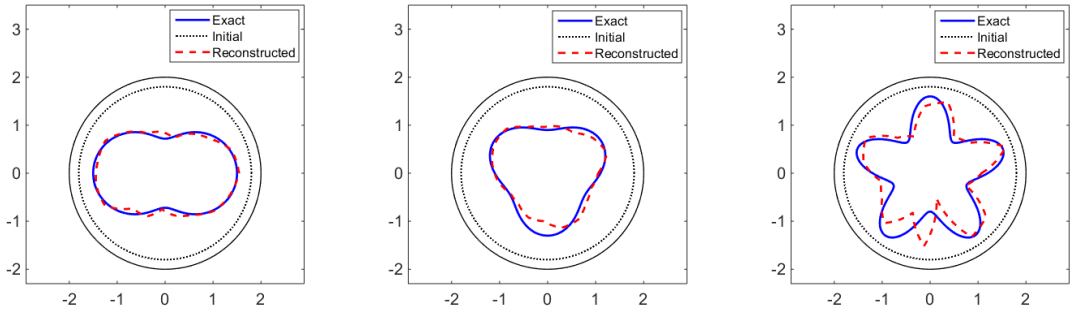
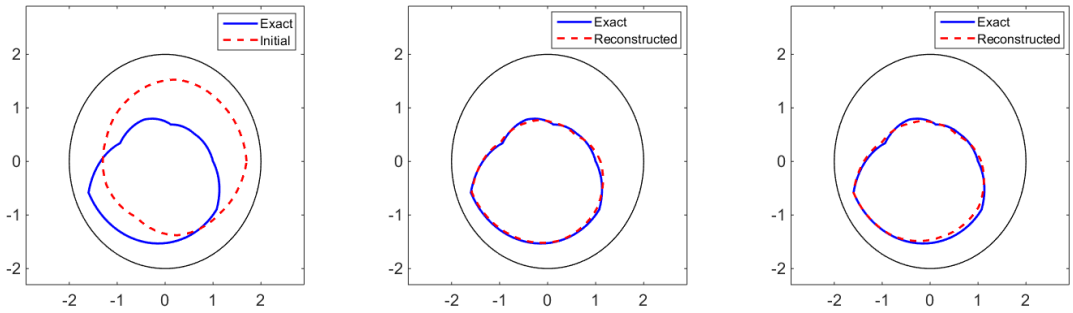


Figure 7: Reconstructions obtained by Algorithm 2 for the geometries discussed in Table 1 but for noisy data with noise level $\epsilon = 5\%$.

4.3 Discussion on the choice of L

As mentioned earlier for a given τ, it is not clear which value of L would lead to potentially best performances of Algorithm 2. We provide here some elementary numerical investigations in the case where the exact geometry is given by the parametrization $\mathcal{T}_n$ (2.3) for given n. In that case it is possible to use a stopping rule related to the accuracy of the reconstruction, e^k defined as $e^k = \frac{\|\overline{R}-R^k\|_2}{\|\overline{R}\|_2}$ where $\Sigma_{\overline{R}}$ is the exact interface.

In the example below we choose $n = 9$, $\overline{R} = (1, 0.8, 0.7, 0.9, 1, 1.7, 1.6, 1.5, 1.4)$ and the initial guess $R^0 = (1.7, 1.6, 1.55, 1.4, 1.3, 1.3, 1.2, 1.4, 1.5)$ (see Figure 8(a)).



(a) Initial guess (b) Algorithm 1 (c) Algorithm 2

Figure 8: Comparison between Algorithm 1 and Algorithm 2 with $L = 1$ to achieve 5% relative accuracy in the case of the geometry given by Figure (a) and for noise free data. The gradient descent parameter $\tau = 0.1$ for both algorithms.

Table 2 gives the number of iterations K need to achieve an accuracy $e^k \simeq 5\%$ for noise free data. We also indicate the virtually total cost $2KL$ that represents the total number of iterations for direct problems. We observe that $L = 1$ provides the lowest values for KL and $L = 2$ or $L = 3$ comparative performances. This shows again the relevance of Algorithm 2 and that a few DDM iterations are sufficient to provide good solution for the inverse problem.

	$\tau = 0.1$		$\tau = 0.05$		$\tau = 0.005$	
L	K	$2KL$	K	$2KL$	K	$2KL$
1	53	106	55	110	359	718
2	29	116	41	164	355	1420
3	29	174	39	234	355	2130
4	30	240	41	328	354	2832
5	30	300	39	390	354	3540

Table 2: Total number of iterations for Algorithm 2 to achieve 5% relative accuracy in the case of the geometry given by Figure 8 and for noise free data.

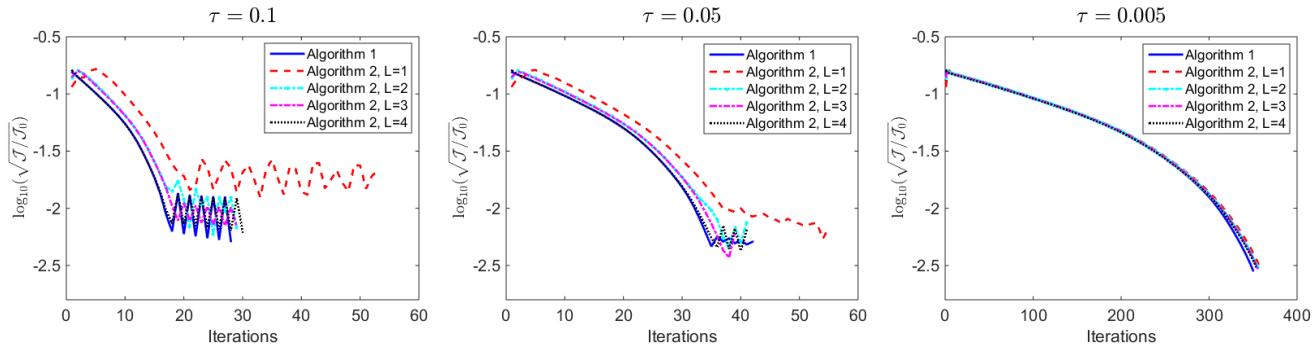


Figure 9: Comparison of the evolution of $\log_{10}(\sqrt{\mathcal{J}/\mathcal{J}_0})$ between Algorithm 1 and Algorithm 2 applied to the example of Figure 8 with different gradient descent parameter $\tau = 0.1$ (left), $\tau = 0.05$ (middle) and $\tau = 0.005$ (right).

References

[1] L. Afraites, M. Dambrine, and D. Kateb. *Shape Methods for the Transmission Problem with a Single Measurement.* Numer. Funct. Anal. Optim., 28(5-6), 519-551, 2007.

[2] L. Afraites, M. Dambrine, and D. Kateb. *On second order shape optimization methods for electrical impedance tomography.* SIAM J. Control Optim., 47(3), 1556-1590, 2008.

[3] A. Aspri, E. Beretta, E. Francini and S. Vessella, *Lipschitz Stable Determination of Polyhedral Conductivity Inclusions from Local Boundary Measurements*, SIAM Journal on Mathematical Analysis, 54, 5, 5182-5222, 2022.

[4] K. Astala and L. Päivärinta. *Calderón's inverse conductivity problem in the plane.* Ann. of Math., 163, 265-299, 2006.

[5] E. Beretta and E. Francini. *Global Lipschitz stability estimates for polygonal conductivity inclusions from boundary measurements* Appl. Anal., 01(10), 35363549, 2022

[6] M. Bonazzoli, H. Haddar, and T. A. Vu. *Convergence analysis of multi-step one-shot methods for linear inverse problems.* [Research Report] RR-9477, Inria Saclay; ENSTA ParisTech, 2022.

[7] L. Borcea. *Electrical impedance tomography.* Inverse Problems, 18(6), R99R136, 2002.

[8] M. Burger and W. Mhlhuber. *Iterative regularization of parameter identification problems by sequential quadratic programming methods.* Inverse Problems, 18(4), 943969, 2002.

[9] S. Chaabane, B. Charfi, and H. Haddar. *Reconstruction of discontinuous parameters in a second order impedance boundary operator.* Inverse Problems, 32(10), 105004, 2016.

[10] S. Chaabane, C. Elhechmi, and M. Jaoua. *A stable recovery algorithm for the Robin inverse problem.* IMACS, Math. Comput. Simul., 66(4-5), 367-383, 2004.

[11] S. Chaabane, I. Feki, and N. Mars. *Numerical reconstruction of a piecewise constant Robin parameter in the two-or three-dimensional case.* Inverse Problems, 28(6), 065016, 2012.

[12] S. Chaabane and M. Jaoua. *Un algorithme d'identification de frontières soumises à des conditions aux limites de Signorini.* M2AN, 34(3), 707-722, 2000.

[13] V. Dolean, P. Jolivet, and F. Nataf. *An Introduction to Domain Decomposition Methods.* SIAM, Philadelphia, 2015.

[14] O. Dubois. *Optimized Schwarz methods for the advection-diffusion equation and for problems with discontinuous coefficients.* PhD thesis. McGill University, 2007.

[15] M. J. Gander. *Optimized Schwarz methods.* SIAM J. Numer. Anal., 44(2), 699-731, 2006.

[16] N. Gauger, A. Griewank, A. Hamdi, C. Kratzenstein, E. zkaya, and T. Slawig. *Automated extension of fixed point PDE solvers for optimal design with bounded retardation.* In: Constrained Optimization and Optimal Control for Partial Differential Equations. International Series of Numerical Mathematics, 160, 99122, Springer Basel, Basel, 2012.

[17] L. Gerardo-Giorda and F. Nataf. *Optimized Schwarz methods for unsymmetric layered problems with strongly discontinuous and anisotropic coefficients.* Technical report, 561, submitted. Ecole Polytechnique, CMAP, CNRS UMR 7641, Paris, France, 2004.

[18] A. Griewank. *Projected Hessians for Preconditioning in One-Step One-Shot Design Optimization.* In: Large-Scale Nonlinear Optimization. Nonconvex Optimization and Its Applications, 83, 151171, Springer US, Boston, MA, 2006.

[19] E. Haber and U. M. Ascher. *Preconditioned all-at-once methods for large, sparse parameter estimation problems.* Inverse Problems, 17(6), 18471864, 2001.

[20] H. Haddar and R. Kress. *Conformal mapping and impedance tomography.* Inverse Problems, 26(7), 074002, 2010.

[21] A. Hamdi and A. Griewank. *Reduced quasi-Newton method for simultaneous design and optimization.* Computational Optimization and Applications, 49(3), 521548, 2009.

[22] M. Hanke, B. Harrach, and N. Hyvnen. *Justification of point electrode models in electrical impedance tomography.* Math. Models Methods Appl. Sci., 21(6), 1395-1413, 2011.

[23] M. Hassine and I. Kallel. *KohnVogelius formulation and topological sensitivity analysis based method for solving geometric inverse problems.* Arab. J. Math. Sci., 24(1), 4362, 2018.

[24] F. Hecht. *New development in FreeFem++.* J. Numer. Math., 20(3-4), 251265, 2012.

[25] F. Hettlich and W. Rundell. *The determination of a discontinuity in a conductivity from a single boundary measurement.* Inverse Problems, 14(1), 67-82, 1998.

[26] V. Isakov. *Inverse Problems for Partial Differential Equations.* Applied Mathematical Sciences, 127, Springer, 2017.

[27] C. Japhet and F. Nataf. *The best interface conditions for domain decomposition methods: Absorbing boundary conditions.* In: Absorbing Boundaries and Layers, Domain Decomposition Methods: Applications to Large Scale Computations, 348-373, Nova Science, New York, 2001.

[28] C. Japhet, F. Nataf, and F. Rogier. *The optimized order 2 method. Application to convection-diffusion problems.* Future Gener. Comput. Sys., 18(1), 17-30, 2001.

[29] B. Kaltenbacher, A. Kirchner, and B. Vexler. *Goal oriented adaptivity in the IRGNM for parameter identification in PDEs II: all-at-once formulations.* Inverse Problems, 30(4), 045002, 2014.

[30] T. A. Khan. *Review on Electrical Impedance Tomography: Artificial Intelligence Methods and its Applications.* Algorithms, 12(5), 88, 2019.

[31] R. V. Kohn and M. Vogelius. *Relaxation of a variational method for impedance computed tomography.* Communications on Pure and Applied Mathematics, 40(6), 745-777, 1987.

[32] P.-L. Lions. *On the Schwarz alternating method. III. A variant for nonoverlapping subdomains.* In: Third International Symposium on Domain Decomposition Methods for Partial Differential Equations. SIAM, 202-223, 1990.

[33] J. L. Mueller and S. Siltanen. *Linear and Nonlinear Inverse Problems with Practical Applications.* SIAM, 2012.