The Dissertation Committee for Benjamin Crestel
certifies that this is the approved version of the following dissertation:

Advanced techniques for multi-source, multi-parameter, and multi-physics inverse problems

Committee:

Omar Ghattas, Supervisor

Tan Bui-Thanh

Bjorn Engquist

Sergey Fomel

Kui Ren

Georg Stadler

Advanced techniques for multi-source, multi-parameter, and multi-physics inverse problems

by

Benjamin Crestel

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2017

Dedicated to Marianne, Anémone, Opaline, and Aramis.

Acknowledgments

This dissertation marks the conclusion of a solitary, at times lonely, journey that would, however, not have been possible without the help and support of many people. I am very grateful to my advisor, Omar Ghattas, who trusted me and accepted me into his research group. His passion and enthusiasm for research were an inspiration throughout my doctoral program. I worked with Georg Stadler since my first day at UT, and I greatly benefited from his advice and guidance. For that, I owe him a debt of gratitude. I wish to thank my committee members for their service, and for sharing their expertise with me. I also thank all the members of the CCGO group that I had the pleasure to meet during these 5 years for the stimulating and friendly discussions; but more specifically, I want to thank Umberto Villa for developing and maintaining the hIPPYlib library which I used in my dissertation, and for keeping his door open at all times, and Johann Rudi who volunteered to proofread the introduction of this dissertation. I must also thank Sue Rodriguez for managing the daily life of the group so brilliantly, and shielding us from most administrative responsibilities. Finally, I could not close this section without thanking my family who patiently waited for me, and coped with my erratic schedule and fluctuating mood.

Advanced techniques for multi-source, multi-parameter, and multi-physics inverse problems

Publication No. _____

Benjamin Crestel, Ph.D.
The University of Texas at Austin, 2017

Supervisor: Omar Ghattas

With the increase in compute power and the advent of the big data era, inverse problems have grown more complex, attempting to extract more information and to use more data. While this evolution manifests itself in multiple forms, we focus in this dissertation on three specific aspects: multi-source, multi-parameter, and multi-physics inverse problems.

The computational cost of solving a multi-source inverse problem increases linearly with the number of experiments. A recently proposed method to decrease this cost uses only a small number of random linear combinations of all experiments for solving the inverse problem. This approach applies to inverse problems where the PDE solution depends linearly on the right-hand side function that models the experiment. As this method is stochastic in essence, the quality of the obtained reconstructions can vary, in particular when only a small number of combinations are used. We propose to replace the random weights traditionally used in the linear combinations of the experiments, with deterministic weights (or, encoding weights). We approach the computation of these weights as an optimal experimental design problem, and

develop a Bayesian formulation for the definition and computation of encoding weights that lead to a parameter reconstruction with the least uncertainty. We call these weights A-optimal encoding weights. Our framework applies to inverse problems where the governing PDE is nonlinear with respect to the inversion parameter field. We formulate the problem in infinite dimensions and follow the optimize-then-discretize approach, devoting special attention to the discretization and the choice of numerical methods in order to achieve a computational cost that is independent of the parameter discretization. We elaborate our method for a Helmholtz inverse problem, and derive the adjoint-based expressions for the gradient of the objective function of the optimization problem for finding the A-optimal encoding weights. The proposed method is potentially attractive for real-time monitoring applications, where one can invest the effort to compute optimal weights offline, to later solve an inverse problem repeatedly, over time, at a fraction of the initial cost.

We define a multi-parameter inverse problem, also called joint inverse problem, as the simultaneous inference of multiple parameter fields. In this dissertation, we concentrate on two types of multi-parameter inverse problems. In the first case, we have at our disposal a single type of observations, generated by a single physical phenomenon which depends on multiple parameters. In the second case, we utilize multiple datasets generated from physical phenomena that depend on different parameters; when the data are generated from different physics, this is a multi-physics inverse problem. The regularization of a multi-parameter inverse problem plays a critical role. It not only acts as a regularizer to the inverse problem, but can also be used to impose coupling between the inversion parameters when they are known to share similar structures. We compare four joint regularizations terms: the cross-gradient, the

normalized cross-gradient, the vectorial total variation, and a novel regularizer based on the nuclear norm of a gradient matrix. Following comprehensive numerical investigations, we concluded that vectorial total variation leads to the best reconstructions. We next devoted our attention to develop an efficient primal-dual Newton solver for joint inverse problems regularized with vectorial total variation. Introducing an auxiliary dual variable in the first-order optimality condition, which we then solve using Newton method, we were able to reduce the nonlinearity in the inverse problem. Through an extensive numerical investigation, we showed that this solver is scalable with respect to the mesh size, the hyperparameter, and the number of inversion parameters. We also observed that it significantly outperforms the classical Newton method and the popular lagged diffusivity method when fine convergence tolerances are needed.

Multi-physics inverse problems are becoming more popular as a way to enhance the quality of the reconstructions by combining the strengths of multiple imaging modalities. In this dissertation, we specialize to the case of full-waveform inversion, and the presence of local minima in its objective function when using high-frequency data. The most practical workaround today remains a continuation scheme over the frequency of the source term. However, in a seismic exploration setting, modern equipment does not allow to generate data of sufficiently low frequencies. One potential application of multi-physics inverse problems is to allow an auxiliary physical phenomenon, e.g., electromagnetic waves, to provide the missing low-frequency information for full-waveform inversion. In this dissertation, we provide supporting evidence for this approach when using the vectorial total variation functional as a regularization.

Table of Contents

List of Tables

List of Figures

Chapter 1

A primer on the numerical solution of large-scale inverse problems constrained by Partial Differential Equations

We start this chapter with an introduction to inverse problems governed by Partial Differential Equations (PDEs), and the computational methods enabling their efficient solution. In section 1.4, we concentrate on full-waveform inversion, a specific example of inverse problems that poses some unique challenges. We close this chapter by introducing three aspects of inverse problems that will be developed in the rest of this dissertation: multi-source, multi-parameter, and multi-physics inverse problems.

1.1 Introduction

Inverse problems are ubiquitous in science and engineering. They arise whenever one attempts to infer parameters m from indirect observations $\mathbf{d}$ and from a mathematical model—the parameter-to-observable map, $\mathcal{F}(\cdot)$—for the physical phenomenon that relates m and $\mathbf{d}$. For some applications, it is common to use observations obtained from different experiments, when available, to improve the quality of the parameter estimation. Suppose N_s experiments are conduced, indexed by $i \in \{1, \ldots, N_s\}$. The i-th experiment results in observations $\mathbf{d}_i$ and the corresponding parameter-to-observable map is denoted by $\mathcal{F}_i(m)$. Following a deterministic approach to this inverse problem results

in the least-squares minimization problem

$$\min_m \left\{ \frac{1}{2N_s} \sum_{i=1}^{N_s} \|\mathcal{F}_i(m) - \mathbf{d}_i\|^2 + \mathcal{R}(m) \right\}, \tag{1.1}$$

where the objective function is the sum of a data-misfit term, and an appropriate regularization operator $\mathcal{R}$ to cope with the ill-posedness of the inverse problem. We discuss the role of the regularization term $\mathcal{R}$ in more details in section 1.2. For PDE-constrained inverse problems, which are the focus of this dissertation, each evaluation of $\mathcal{F}_i(m)$ entails the solution u_i of a PDE, i.e., $\mathcal{A}(m; u_i) = f_i$, and this u_i is then restricted by an observation operator B to a subset of the domain where observations are available, i.e., $\mathcal{F}_i(m) = Bu_i$. We do not make any assumptions on the PDE, and the operator $\mathcal{A}$ could be nonlinear in m and/or u_i, as well as time-dependent. The norm $\|\cdot\|$ for the data-misfit term in (1.1) is dictated by the type of observations we have. In the case of a finite number of observations, i.e., $\mathbf{d}_i \in \mathbb{R}^q$, this norm is the Euclidean norm. In the case of B being the identity operator and $u_i \in L^2(\Omega)$, with $\Omega \subset \mathbb{R}^d$ the physical domain, we choose the L^2-norm. In the next section, we also show an example where B is a trace operator.

1.1.1 Examples of inverse problems

We now illustrate the framework (1.1) with a few examples of inverse problems and their formulations.

Imaging applications Denoising and deblurring are two imaging techniques used to reconstruct a picture that was either polluted by noise, or was blurred out. This is a central problem for the field of imaging science, and this problem can be formulated as an inverse problem [86]; albeit a simpler one as no PDE

is involved. We denote by K the operator that corrupted the original picture; in the case of a denoising application K is the identity operator, whereas in the case of a deblurring problem K corresponds to the blurring kernel. Given a blurry or noisy image $\mathbf{d} \in \mathbb{R}^{n\times n}$ made of n^2 pixels, we look for a reconstruction u_d by solving the minimization problem

$$\mathbf{u}_d \coloneqq \min_{\mathbf{u}\in\mathbb{R}^{n\times n}} \left\{ \frac{1}{2}|K\mathbf{u} - \mathbf{d}|^2 + \mathcal{R}(\mathbf{u}) \right\}. \tag{1.2}$$

This is an example of a linear inverse problem, as the parameter-to-observable map, given by $\mathcal{F}(u) = Ku$, is linear with respect to the inversion parameter u.

Porous media flow A more complex example of imaging technique targets geological properties of a domain using observation of a subsurface flow [69]. Using pressure information at different locations of the domain, one can attempt to reconstruct the log-permeability m_d of the subsurface. If the flow reached a steady state, its pressure can be described by a Poisson equation with an heterogeneous coefficient. Denoting the pressure in the subsurface by u, the inverse problem is defined by

$$\begin{gathered} m_d \coloneqq \min_{m\in L^2(\Omega)} \left\{ \frac{1}{2}|Bu - \mathbf{d}|^2 + \mathcal{R}(m) \right\} \\ \text{where,} \\ \begin{cases} -\nabla\cdot\left(e^m \nabla u\right) = f, & \text{in } \Omega \\ e^m \nabla u \cdot n = g, & \text{on } \partial\Omega_n \\ u = h, & \text{on } \partial\Omega_d, \end{cases} \end{gathered} \tag{1.3}$$

where the boundary of the physical domain Ω is split into $\partial\Omega = \partial\Omega_n \cup \partial\Omega_d$. The observation operator in that case could be a pointwise observation, that is, $Bu = [u(x_1), u(x_2), \ldots, u(x_q)]^T$. This requires the solution u to be continuous, which in turn requires sufficient regularity of the source term f. If

that is not the case, Bu can be defined by convolutions between the solution u and smooth compactly supported functions ϕ_i that are centered at different locations of the domain. The inverse problem (1.3) is nonlinear. Denoting the PDE constraint in (1.3) by $\mathcal{A}(m)u = f$, we can write the parameter-to-observable map as $\mathcal{F}(m) = Bu = B[\mathcal{A}(m)]^{-1}f$, which depends nonlinearly on the log-permeability m. Not all PDE constrained inverse problems are nonlinear, however. A closely related example would be to assume m to be known, and instead attempt to recover the source term f. In that case, the parameter-to-observable map becomes $\mathcal{F}(f) = B[\mathcal{A}(m)]^{-1}f$, which is a linear operator of the source term. An example of such an inverse problem is the reconstruction of a heat source that is not directly observable, using temperature measurements away from the source.

Transport problem Other quantities of interest besides a medium parameter or a source term can be inverted for. An example of inversion for the initial conditions of a PDE is to reconstruct the original concentration of a gas displacing through the air, given recordings of its concentration at different locations and times [36]. The propagation of the gas follows a convection-diffusion equation. The formulation for that type of inverse problem is given by

$$\bar{u}_0 \coloneqq \min_{u_0} \left\{ \frac{1}{2} \int_0^T |Bu(t) - \mathbf{d}(t)|^2 \, dt + \mathcal{R}(u_0) \right\}$$

where,

$$\begin{cases} u_t - k\Delta u + \mathbf{v}\cdot\nabla u = 0, & \text{in } \Omega \times (0,T), \\ u = u_0, & \text{in } \Omega \times \{t = 0\}, \\ k\nabla u \cdot \mathbf{n} = 0, & \text{on } \partial\Omega_n \times (0,T), \\ u = 0, & \text{on } \partial\Omega_d \times (0,T). \end{cases}$$

Another specificity of that example is the time dependence of the PDE operator and the observations. This plays a subtle role in the discretization of the inverse problem; we discuss this issue in section 1.4.

Creeping flow of ice sheets Our last example is the inversion for a friction coefficient in the boundary conditions of a creeping ice sheet, given displacements at the surface of the ice sheet [55]. Precise knowledge of the sliding rate of the ice over the bedrock is key to better understand the movement of large ice sheets. This sliding at the base is modeled by a Robin boundary condition, parametrized by the inversion parameter β. The corresponding inverse problem is given by

$$\beta_d \coloneqq \frac{1}{2}\int_{\Gamma_t} \frac{|Bu-d|}{|d|^2+\varepsilon}\,ds + \mathcal{R}(\beta)$$

where,

$$\begin{cases} -\nabla\cdot\left[\eta(u)(\nabla u+\nabla u^T)-Ip\right]=\rho g, & \text{in } \Omega,\\ \nabla\cdot u = 0, & \text{in } \Omega,\\ \sigma n = 0, & \text{on } \Gamma_t,\\ u\cdot n = 0,\ T\sigma_u n + e^{\beta} T u = 0, & \text{on } \Gamma_b, \end{cases}$$

and some additional constitutive relations that are omitted here for brevity. Another specificity of that example is that the observation operator is a trace operator, and the data-misfit is integrated over a part of the boundary of the domain Γ_t.

1.2 Regularization functional

Except in some very rare cases, like the denoising or deblurring applications, the minimization of the data-misfit term alone in (1.1) is an ill-posed

problem. A problem can fail to be well-posed if it doesn't have a solution, if the solution is not unique, or if the solution doesn't depend continuously on the data given in the problem [33]. Formulation (1.1) is generally ill-posed because it does not have a unique solution. This is due to the limited number of observations and the relative insensitivity of the observations to some perturbations in the inversion parameter. The mathematical solution to that problem is to regularize the ill-posed formulation. There exist multiple ways to do so [21, 86], but in this dissertation, we concentrate on penalization techniques that regularize formulation (1.1) by adding a penalty term $\mathcal{R}(m)$ to the objective function. This regularization term is chosen to eliminate the solutions the parameter-to-observable is insensitive to. This approach is preferred for large-scale inverse problems, as it facilitates a matrix-free approach to the numerical solutions of the inverse problem (1.1) (see section 1.3). We now introduce two popular regularization terms among penalization techniques for inverse problems, the Tikhonov regularization and the total variation (TV) regularization.

1.2.1 Tikhonov regularization

The Tikhonov regularization is given, for $m \in H^1(\Omega)$, by

$$\mathcal{R}(m) = \frac{1}{2} \int_\Omega (\alpha |m|^2 + \beta |\nabla m|^2)\, dx,$$

where $\alpha, \beta \geq 0$, and $(\alpha, \beta) \neq (0, 0)$. In the most common case of $\beta > 0$, the Tikhonov functional will penalize oscillatory solutions of the inverse problem, and the reconstructed parameter will be smooth, with $m \in H^1(\Omega)$. It is a popular choice of regularization in the literature. The main reasons for that popularity are: (1) the Tikhonov functional is quadratic, leading to simple

derivatives; (2) its Hessian (second derivative) is the sum of the identity operator and the Laplacian, making the use of Newton's method to solve the inverse problem simpler (see section 1.3); and (3) the question of choosing coefficients α, β has been extensively studied in the literature. We mention two popular methodologies for that selection process. If the noise level in the observations $\mathbf{d}$ is known, the Discrepancy Principle selects the regularization coefficients that leads to a data misfit residual, at the minimizer, of the same magnitude as the noise level [66]. With less assumptions on the noise in the observations, the L-curve proposes a geometric approach to select regularization parameters leading to a good trade-off between under- and over-regularization [49].

However, for certain applications, particularly in seismic or medical imaging, the reconstructed parameter field is expected to present discontinuities across hypersurfaces, a feature that Tikhonov regularization cannot preserve [18]. This is the main motivation to use the total variation regularization.

1.2.2 Total variation regularization

The total variation regularization was developed inside the imaging community [74] in an attempt to obtain reconstructions displaying sharp edges. It is a non-smooth convex functional, often presented in the form

$$\mathcal{R}_{\text{TV}}(m) = \int_{\Omega} |\nabla m| \, dx. \tag{1.4}$$

Although formulation (1.4) is the most commonly used for computational applications, it is only well-defined for reconstructions $m \in W^{1,1}(\Omega)$ that are almost surely continuous functions [18]. An alternative definition that requires

less regularity is given by (1.5), for $m \in L^1(\Omega)$,

$$\mathcal{R}_{\text{TV}}(m) = \sup \left\{ -\int_\Omega m \, \text{div}\phi \, dx; \; \phi \in C_c^\infty(\Omega; \mathbb{R}^d), \; |\phi(x)| \leq 1, \; \forall x \in \Omega \right\}, \quad (1.5)$$

where $\Omega \subset \mathbb{R}^d$ is the physical domain. The functions that verify $\mathcal{R}_{\text{TV}}(m) < +\infty$ in (1.5) are said to have bounded variations, and form the space $BV(\Omega)$. This space, equipped with the norm $\|m\|_{BV} \coloneqq \mathcal{R}_{\text{TV}}(m) + \|m\|_{L^1}$, forms a Banach space. The space $BV(\Omega)$ is much larger than $W^{1,1}(\Omega)$, and contains functions with jump-discontinuities or edges [20].

A fundamental property of TV that helps understand its effect on the solution to (1.1) is the co-area formula. It shows that the total variation of a function is also the accumulated surfaces of all its level sets, which translates into the formula,

$$\mathcal{R}_{\text{TV}}(m) = \int_{-\infty}^{+\infty} \text{Per}(\{m > s\}) \, ds,$$

with the perimeter of a set E, $\text{Per}(E)$, defined as the total variation of its characteristic function [18]. A characteristic feature of reconstructions obtained using the TV functional is their tendency to be piecewise constant; a phenomenon known as staircasing effect. The explanation of that phenomenon is a complex matter, not fully understood in the general case, but seems to be related to the co-area formula [56]. Despite the many advantages of the TV regularization, its use in PDE-constrained inverse problems presents some challenges.

1.2.2.1 Modified TV

The total variation functional is highly nonlinear, and non-differentiable when $|\nabla m| = 0$. The non-differentiability of the TV functional is often addressed, in the inverse problem community, by approximating the L^1 norm

in (1.4). Two popular approximations are the modified TV,

$$\mathcal{R}_{\mathrm{TV},\varepsilon}(m) \coloneqq \int_\Omega \sqrt{|\nabla m|^2 + \varepsilon}\, dx, \tag{1.6}$$

with $\varepsilon > 0$, and the Huber switching function, $\tilde{\mathcal{R}}_{\mathrm{TV}}(m) \coloneqq \varphi(|\nabla m|)$, where

$$\varphi(t) = \begin{cases} t & \text{, when } t \geq \varepsilon, \\ \dfrac{t^2}{2\varepsilon} + \dfrac{\varepsilon}{2}, & \text{when } t < \varepsilon. \end{cases}$$

Although an approximation to the square root regularizes the TV functional, it also reduces the staircasing phenomenon by smearing the sharp discontinuities. This approximation to the square root therefore introduces a trade-off between sharp edges in the reconstruction and rapid convergence of the inverse problem.

To accommodate the scale of a specific inverse problem (e.g., noise level in the data, size of the domain, type of misfit function), we also introduce a regularization parameter $\gamma > 0$ in the definition of the inverse problem,

$$\min_m \left\{ \frac{1}{2}|\mathcal{F}(m) - \mathbf{d}|^2 + \gamma \int_\Omega |\nabla m|\, dx \right\}.$$

Over the years, several methodologies have been developed to automatically select the constants ε and γ (see for instance [29]). Unlike in the case of Tikhonov regularization, none of these methodologies gained sufficient traction, and to this day, the values of these constants are generally selected empirically (see for instance [6]).

1.2.2.2 Numerical solution

A single modification of the TV functional alone is not sufficient to obtain good computational performance, and the numerical solution of inverse problems regularized with TV also requires the use of specialized numerical

methods. In section 4 we introduce a primal-dual Newton method for joint inverse problems regularized with vectorial total variation; it generalizes existing second-order methods for inverse problems regularized with TV. Two methods help illustrate the primal-dual Newton method; the Augmented Lagrangian method, and the primal-dual method. We spend the rest of this section discussing both approaches.

Augmented Lagrangian approach The Augmented Lagrangian approach [67], also connected to the Split Bregman method [39] and Alternating directions method of multipliers (ADMM) [13], consists of relaxing the computationally challenging term, here the L^1 norm of the gradient of the parameter, by introducing an auxiliary variable. In the case of an inverse problem regularized by TV, we would replace the initial formulation,

$$\min_m \left\{ \frac{1}{2}|\mathcal{F}(m) - \mathbf{d}|^2 + \gamma \int_\Omega |\nabla m| \, dx \right\},$$

with

$$\min_{\substack{(m,p) \\ p=\nabla m}} \left\{ \frac{1}{2}|\mathcal{F}(m) - \mathbf{d}|^2 + \gamma \int_\Omega |p| \, dx \right\}.$$

We next write down a Lagrangian for that constrained minimization problem, augmenting the Lagrangian with a penalty term,

$$\mathscr{L}(m, p, \lambda) = \frac{1}{2}|\mathcal{F}(m) - \mathbf{d}|^2 + \gamma \int_\Omega |p| \, dx + (\lambda, p - \nabla m) + \frac{\alpha}{2}\|p - \nabla m\|^2.$$

A minimizer is then computed iteratively. At every step, one alternates a minimization step over m, a minimization step over p, and a gradient ascent

step for the Lagrange multiplier, i.e.,

$$\begin{aligned} m^{k+1} &= \arg\min_m \mathscr{L}(m, p^k, \lambda^k), \\ p^{k+1} &= \arg\min_p \mathscr{L}(m^{k+1}, p, \lambda^k), \\ \lambda^{k+1} &= \lambda^k + \alpha(p^{k+1} - \nabla m^{k+1}). \end{aligned}$$

Primal-dual approach The primal-dual approach hinges on the concept of duality, a central concept of convex optimization [72]. Let us consider a functional $F : X \to \mathbb{R}$, defined on a Hilbert space X. Its Legendre-Fenchel conjugate, at a point $p \in X$, is defined by

$$F^*(p) = \sup_{x \in X} \langle p, x \rangle - F(x).$$

A central result of convex analysis is that for functionals $F \in \Gamma^0(X)$, i.e., for functionals that are convex, proper, and lower-semicontinuous, we have $F^{**}(p) = F(p)$. Let us now define the prototypical imaging problem

$$\min_{x \in X} F(Ax) + G(x), \tag{1.7}$$

where $F \in \Gamma^0(Y)$, Y a Hilbert space, $G : X \to \mathbb{R}$ a convex functional, and $A : X \to Y$ a linear operator; for instance, with $F(y) = \int_\Omega |y| \, dx$, $Ax = \nabla x$, and $G(x) = |x - d|^2/2$, we recover the denoising problem (1.2) regularized with TV. Since $F \in \Gamma^0(Y)$, we can re-write the first term in (1.7) using the Legendre-Fenchel transformation, to obtain the primal-dual form (1.8),

$$\min_{x \in X} \max_{y \in Y} \langle y, Ax \rangle - F^*(y) + G(x). \tag{1.8}$$

The idea of the primal-dual approach consists of replacing the solution of problem (1.7) with the iterative solution of problem (1.8), by alternating between

a gradient descent in x and a gradient ascent in y. The functionals F and G being potentially non-differentiable, the gradient descent (resp. ascent) step is replaced by a proximal gradient descent (resp. ascent) step. The proximal operator, $(I + \gamma\partial F)^{-1}$, is defined as

$$(I + \gamma\partial F)^{-1}(x) \coloneqq \arg\min_{p \in X} \frac{1}{2}\|p - x\|^2 + \gamma F(p).$$

The complete primal-dual algorithm to solve problem (1.8) is then given by

$$y^{k+1} = (I + \sigma\partial F^*)^{-1}(y^k + \sigma A x^k),$$
$$x^{k+1} = (I + \tau\partial G)^{-1}(x^k - \tau A^* y^{k+1}),$$

where $\sigma, \tau > 0$. In the case of the denoising application (1.2), the Legendre-Fenchel conjugate of F is given by the characteristic function of the set $K \coloneqq \left\{p \in X;\, p = -\nabla\cdot\xi,\, \xi \in X^d,\, \sum_{i=1}^d \xi_i(x)^2 \leq 1,\, \forall x \in \Omega\right\}$, i.e., $F^*(p) = 0$ if $p \in K$, and $F^*(p) = +\infty$ if $p \notin K$. The proximal operator for F^* is then an orthogonal projection onto the set K, a rather simple step to compute.

1.3 Numerical solution of large-scale inverse problems

The solution of inverse problem (1.1) typically requires the use of iterative methods. Iterative methods require the availability of the first (and ideally, also second) derivative of the objective function with respect to the parameter field m [67, 86]. In this section, we discuss the computation of these derivatives, and methods from numerical optimization that can be utilized to solve large-scale nonlinear inverse problems constrained by PDEs.

1.3.1 Derivatives of the objective function

Without loss of generality, let us assume the observation operator B is a pointwise observation operator, and let us call $J(\{u_i\}_i, m)$ the objective

function of the inverse problem (1.1), i.e.,

$$J(\{u_i\}_i, m) \coloneqq \frac{1}{2N_s} \sum_{i=1}^{N_s} |Bu_i - \mathbf{d}_i|^2 + \mathcal{R}(m), \tag{1.9}$$

where each u_i, for $i = 1, \ldots, N_s$, solves the PDE $\mathcal{A}(m, u_i) = f_i$. There exist two approaches to compute discrete derivatives of the objective functional (1.9). One can discretize the objective function first, then compute derivatives of the discrete functional; this approach is called discretize-then-optimize (DTO). DTO is guaranteed to produce a gradient that is consistent with the discrete objective functional. However, the calculations for the derivatives of the discrete objective functional are often tedious, and the definition of the adjoint equation (see later in this section) is sometimes unintuitive. In the alternative approach, optimize-then-discretize (OTD), we formulate the optimization problem in function space, then discretize that infinite-dimensional optimization problem. The calculations are often easier to carry out, but the discrete derivatives are not guaranteed to be consistent with the discrete objective function (see [88] for an example where the OTD approach requires a correction). In the rest of this dissertation, the OTD approach is the method of choice, and we spend the rest of this section reviewing the main techniques to compute derivatives of J with respect to the parameter field m [26, 51].

1.3.1.1 First derivative

Sensitivity approach Let us assume that the PDE operator $\mathcal{A}$ is continuously Fréchet differentiable. We can then apply the implicit function theorem to define a solution operator for that PDE. This solution operator maps the parameter m onto a unique solution $u_i(m)$. Replacing u_i with $u_i(m)$ in (1.9) leads to the reduced unconstrained optimization problem, $\min_m \{J(\{u_i(m)\}_i, m)\}$.

Let us also assume that the objective function J and all solution operators $u_i(\cdot)$ are continuously Fréchet differentiable. We denote by J_m (resp. J_{u_i}) the partial derivative of the objective function J with respect to the parameter field m (resp. u_i). We can then apply the chain-rule to differentiate the objective function in a direction $\tilde{m}$,

$$\begin{aligned}\frac{dJ}{dm}(\{u_i(m)\}_i, m)\tilde{m} &= \sum_{i=1}^{N_s} J_{u_i}(\{u_i(m)\}_i, m)u_i'(m)\tilde{m} + J_m(\{u_i(m)\}_i, m)\tilde{m} \\ &= \frac{1}{N_s}\sum_{i=1}^{N_s}(Bu_i(m) - \mathbf{d}_i, Bu_i'(m)\tilde{m}) + \mathcal{R}'(m)\tilde{m}. \quad (1.10)\end{aligned}$$

We can find the expression for the directional derivative of the i^{th} solution operator, $u_i'(m)\tilde{m}$, by differentiating the forward problem, $\mathcal{A}(m, u_i(m)) = f_i$, with respect to the parameter field m, in a direction $\tilde{m}$. This gives

$$\mathcal{A}_m(m, u_i(m))\tilde{m} + \mathcal{A}_u(m, u_i(m))u_i'(m)\tilde{m} = 0.$$

Assuming $\mathcal{A}_u$ is invertible at $(m, u_i(m))$, we can write the sensitivity of the i^{th} solution operator as

$$u_i'(m)\tilde{m} = -\left[\mathcal{A}_u(m, u_i(m))\right]^{-1}\mathcal{A}_m(m, u_i(m))\tilde{m}. \quad (1.11)$$

Plugging expression (1.11) into (1.10), we obtain the expression for the directional derivative of the objective function (1.9) in a direction $\tilde{m}$,

$$\begin{aligned}\frac{dJ}{dm}(\{u_i(m)\}_i, m)\tilde{m} &= \mathcal{R}'(m)\tilde{m} \\ &\quad - \frac{1}{N_s}\sum_{i=1}^{N_s}(Bu_i(m) - \mathbf{d}_i, B\left[\mathcal{A}_u(m, u_i(m))\right]^{-1}\mathcal{A}_m(m, u_i(m))\tilde{m}). \quad (1.12)\end{aligned}$$

We derived expression (1.12) following the sensitivity approach. This approach is rarely used today in reason of its unfavorable computational cost.

Indeed, following discretization of (1.12), the computation of the gradient requires the evaluation of the directional derivative of the objective functional in as many directions as the dimension of the parameter space, i.e., one would need to evaluate $\frac{dJ}{dm}(\{u_i(m)\}_i, m)e_k$ for all basis functions e_k of the discrete parameter space. Since each evaluation requires to solve a PDE, $[\mathcal{A}_u(m, u_i(m))]^{-1}\mathcal{A}_m(m, u_i(m))e_k$, the computational cost of the gradient, for large-scale applications where the PDE operator cannot be factored, becomes prohibitively expensive.

Adjoint approach The adjoint approach avoids the repeated solution of equation (1.11) by taking the adjoint of the operator $[\mathcal{A}_u(m, u_i(m))]^{-1}$ in (1.12). We then write the directional derivative of the objective functional as,

$$\frac{dJ}{dm}(\{u_i(m)\}_i, m)\tilde{m} = \mathcal{R}'(m)\tilde{m} - \frac{1}{N_s}\sum_{i=1}^{N_s}([\mathcal{A}_u(m, u_i(m))]^{-*}B^*(Bu_i(m) - \mathbf{d}_i), \mathcal{A}_m(m, u_i(m))\tilde{m}). \quad (1.13)$$

The computational advantage of formulation (1.13) is best illustrated by introducing the adjoint variables $\{p_i\}_{i=1}^{N_s}$, that each solve the following adjoint equation,

$$[\mathcal{A}_u(m, u_i(m))]^* p_i = -B^*(Bu_i(m) - \mathbf{d}_i). \quad (1.14)$$

The directional derivative of (1.9) becomes

$$\frac{dJ}{dm}(\{u_i(m)\}_i, m)\tilde{m} = \frac{1}{N_s}\sum_{i=1}^{N_s}(p_i, \mathcal{A}_m(m, u_i(m))\tilde{m}) + \mathcal{R}'(m)\tilde{m}. \quad (1.15)$$

The number of PDE solves required to compute the gradient of (1.9), when using the adjoint approach, is independent of the discretization of the parameter space. It requires N_s PDE solves to compute the state variables $\{u_i(m)\}_i$, and

N_s PDE solves to compute the adjoint variables $\{p_i\}_i$, for a total of $2N_s$ PDE solves.

Lagrangian approach The last technique we discuss in this section is the Lagrangian approach [82]. It provides a clear, systematic way to compute derivatives with the adjoint approach. Let us define a Lagrangian by augmenting the objective functional (1.9) with the N_s PDE constraints, in weak form,

$$\mathscr{L}(\{u_i\}_i, \{p_i\}_i, m) \coloneqq \frac{1}{2N_s} \sum_{i=1}^{N_s} |Bu_i - \mathbf{d}_i|^2 + \mathcal{R}(m) + \frac{1}{N_s} \sum_{i=1}^{N_s} (p_i, \mathcal{A}(m, u_i) - f_i)\,. \tag{1.16}$$

The factor $1/N_s$ in front of the PDE constraints in (1.16) is not a requirement of the Lagrangian approach, but simply a convenient re-scaling we adopt. Note that the variables $\{u_i\}_i$ in (1.16) are not necessarily solutions to the PDE constraints; all variables in the Lagrangian are treated, a priori, as independent. The first-order optimality condition of the inverse problem is given by the system of equations,

$$\begin{cases} \mathscr{L}_{u_i}(\{u_i\}_i, \{p_i\}_i, m)\tilde{u} = 0, & \forall \tilde{u}, \\ \mathscr{L}_{p_i}(\{u_i\}_i, \{p_i\}_i, m)\tilde{p} = 0, & \forall \tilde{p}, \\ \mathscr{L}_{m}(\{u_i\}_i, \{p_i\}_i, m)\tilde{m} = 0, & \forall \tilde{m}. \end{cases}$$

One can either tackle that optimization problem through a full-scale approach, by solving for all variables $(\{u_i\}_i, \{p_i\}_i, m)$ at the same time [10, 11], or one can adopt a reduced-scale approach. In the latter case, one first solves $\mathscr{L}_{p_j}(\{u_i\}_i, \{p_i\}_i, m) = 0$ to obtain u_j, i.e.,

$$\mathcal{A}(m, u_j) - f_j = 0, \quad \forall j = 1, \ldots, N_s.$$

This corresponds to solving the N_s PDE constraints, and results in setting $u_j = u_j(m)$. The second step is to solve the equations $\mathscr{L}_{u_j}(\{u_i(m)\}_i, \{p_i\}_i, m) = 0$ to obtain p_j, i.e.,

$$B^*(Bu_j(m) - \mathbf{d}_j) + [\mathcal{A}_u(m, u_j(m))]^* p_j = 0, \forall j = 1, \ldots, N_s.$$

We recognize this to be the adjoint equations (1.14). The final step is to plug the quantities computed in the first two steps, the solutions to the constraints $\{u_i(m)\}_i$ and the adjoint variables $\{p_i\}_i$, into $\mathscr{L}_m$. We then obtain the directional derivative (1.15),

$$\mathscr{L}_m(\{u_i(m)\}_i, \{p_i\}_i, m)\tilde{m} = \mathcal{R}'(m)\tilde{m} + \frac{1}{N_s}\sum_{i=1}^{N_s}(p_i, \mathcal{A}_m(m, u_i(m))\tilde{m}).$$

1.3.1.2 Second derivative

We now compute the second-order derivative of the objective function (1.9), first using the sensitivity approach, then using the Lagrangian method with a reduced-space approach.

Sensitivity approach Assuming the PDE operator $\mathcal{A}$, the objective function J, and the solution operators $u_i(m)$ are all twice continuously Fréchet differentiable, we can compute the action of the Hessian of the objective functional (1.9) in a direction $\hat{m}$. Following the sensitivity approach, we obtain

$$\begin{aligned}\frac{d^2J}{dm^2}(\{u_i(m)\}_i, m)(\tilde{m}, \hat{m}) = \frac{1}{N_s}\sum_{i=1}^{N_s}[&(Bu_i'(m)\tilde{m}, Bu_i'(m)\hat{m}) \\ &+(Bu_i(m) - \mathbf{d}, Bu_i''(m)(\tilde{m}, \hat{m}))] + \mathcal{R}''(m)(\tilde{m}, \hat{m}). \quad (1.17)\end{aligned}$$

The only unknown quantities in (1.17) are $u_i''(m)(\tilde{m}, \hat{m})$. We find those quantities by computing the second derivative of the PDE constraints with respect

to m, i.e.,

$$\mathcal{A}_{mm}(m, u_i(m))(\tilde{m}, \hat{m}) + \mathcal{A}_{mu}(m, u_i(m))(\tilde{m}, u_i'(m)\hat{m}) + \mathcal{A}_{um}(m, u_i(m))(u_i'(m)\tilde{m}, \hat{m}) + \mathcal{A}_{uu}(m, u_i(m))(u_i'(m)\tilde{m}, u_i'(m)\hat{m}) + \mathcal{A}_u(m, u_i(m))u_i''(m)(\tilde{m}, \hat{m}) = 0.$$

This leads to expression (1.18) for $u_i''(m)(\tilde{m}, \hat{m})$,

$$u_i''(m)(\tilde{m}, \hat{m}) = -\left[\mathcal{A}_u(m, u_i(m))\right]^{-1}\big[\mathcal{A}_{mm}(m, u_i(m))(\tilde{m}, \hat{m}) + \mathcal{A}_{mu}(m, u_i(m))(\tilde{m}, u_i'(m)\hat{m}) + \mathcal{A}_{um}(m, u_i(m))(u_i'(m)\tilde{m}, \hat{m}) + \mathcal{A}_{uu}(m, u_i(m))(u_i'(m)\tilde{m}, u_i'(m)\hat{m})\big]. \quad (1.18)$$

We now plug expressions (1.18) and (1.14) into expression (1.17) to get the second-derivative of J,

$$\frac{d^2J}{dm^2}(\{u_i(m)\}_i, m)(\tilde{m}, \hat{m}) = \mathcal{R}''(m)(\tilde{m}, \hat{m}) + \frac{1}{N_s}\sum_{i=1}^{N_s}\big[(Bu_i'(m)\hat{m}, Bu_i'(m)\tilde{m}) \\ (p_i, \mathcal{A}_{mm}(m, u_i(m))(\tilde{m}, \hat{m})) + (p_i, \mathcal{A}_{mu}(m, u_i(m))(\tilde{m}, u_i'(m)\hat{m})) + \\ (p_i, \mathcal{A}_{um}(m, u_i(m))(u_i'(m)\tilde{m}, \hat{m})) + (p_i, \mathcal{A}_{uu}(m, u_i(m))(u_i'(m)\tilde{m}, u_i'(m)\hat{m}))\big]. \quad (1.19)$$

Lagrangian approach In the Lagrangian approach, we assume that each variable u_i (resp. p_i) solves its corresponding state (resp. adjoint) equation, i.e., $u_i = u_i(m)$ (resp. p_i solves (1.14)). We next proceed by introducing auxiliary variables, the incremental state variables $\{\hat{u}_i\}_i$ and the incremental adjoint variables $\{\hat{p}_i\}_i$, respectively solutions to the incremental state equations and the incremental adjoint equations. In the first step of the Lagrangian approach, we compute the incremental state variables $\hat{u}_i$, for each $i \in \{1, \ldots, N_s\}$, by

solving the incremental forward equations corresponding to the variation of the forward equation with respect to all variables, i.e.,

$$(\tilde{p}, \mathcal{A}_m(m, u_i(m))\hat{m}) + (\tilde{p}, \mathcal{A}_u(m, u_i(m))\hat{u}_i) = 0.$$

In strong form, this gives the following expression for the incremental state variable $\hat{u}_i$,

$$\begin{aligned} \hat{u}_i &= -\left[\mathcal{A}_u(m, u_i(m))\right]^{-1} \mathcal{A}_m(m, u_i(m))\hat{m}, \\ &= u_i'(m)\hat{m}. \end{aligned} \tag{1.20}$$

The last equality in (1.20) comes from (1.11). The i^{th} incremental state variable $\hat{u}_i$ is therefore the sensitivity of the i^{th} solution operator with respect to the parameter field m, in a direction $\hat{m}$. The second step of the Lagrangian approach consists of computing the incremental adjoint variable $\hat{p}_i$ by solving the incremental adjoint equation, i.e.,

$$\begin{aligned} (B\hat{u}_i, B\tilde{u}) + (\hat{p}_i, \mathcal{A}_u(m, u_i(m))\tilde{u}) + (p_i, \mathcal{A}_{um}(m, u_i(m))(\tilde{u}, \hat{m})) \\ + (p_i, \mathcal{A}_{uu}(m, u_i(m))(\tilde{u}, \hat{u})) = 0. \end{aligned}$$

We can then write $\hat{p}_i$, in strong form, as

$$\begin{aligned} \hat{p}_i = -\left[\mathcal{A}_u(m, u_i(m))\right]^{-*} \big(&B^* B\hat{u}_i \\ &+ \left[\mathcal{A}_{um}(m, u_i(m))(\cdot\,, \hat{m})\right]^* p_i + \left[\mathcal{A}_{uu}(m, u_i(m))(\cdot\,, \hat{u})\right]^* p_i\big). \end{aligned} \tag{1.21}$$

Finally, the action of the Hessian in a direction $\hat{m}$ is given by the variation of $\mathscr{L}_m$, with respect to all variables, in the direction $(\{\hat{u}_i\}_i, \{\hat{p}_i\}_i, \hat{m})$, i.e.,

$$\begin{aligned} \mathcal{R}''(m)(\tilde{m}, \hat{m}) + \frac{1}{N_s}\sum_{i=1}^{N_s} \big[&(\hat{p}_i, \mathcal{A}_m(m, u_i(m))\tilde{m}) \\ &+ (p_i, \mathcal{A}_{mm}(m, u_i(m))(\tilde{m}, \hat{m})) + (p_i, \mathcal{A}_{mu}(m, u_i(m))(\tilde{m}, \hat{u}_i)\big]. \end{aligned} \tag{1.22}$$

Plugging the expressions for the incremental state variables $\hat{u}_i$ (1.20), and the incremental adjoint variables $\hat{p}_i$ (1.21) into expression (1.22), we obtain

$$\mathcal{R}''(m)(\tilde{m},\hat{m}) + \frac{1}{N_s}\sum_{i=1}^{N_s}\big[(Bu_i'(m)\hat{m}, Bu_i'(m)\tilde{m}) + (p_i, \mathcal{A}_{um}(m,u_i(m))(u_i'(m),\hat{m})) + (p_i, \mathcal{A}_{uu}(m,u_i(m))(u_i'(m),u_i'(m)\hat{m})) + (p_i, \mathcal{A}_{mm}(m,u_i(m))(\tilde{m},\hat{m})) + (p_i, \mathcal{A}_{mu}(m,u_i(m))(\tilde{m},u_i'(m)\hat{m})\big],$$

which corresponds to expression (1.19), found using the sensitivity approach.

1.3.2 Numerical optimization techniques

We saw in section 1.3.1 that inverse problems of the form (1.1) can be formulated as nonlinear unconstrained optimization problems, for which the derivatives are readily available using adjoint methods. The numerical solution of such problems can be handled using line search methods, or trust-region methods [67]. We restrict our attention to the former, as this is the method of choice in the rest of this dissertation.

At the k^{th} iteration of a line search method, one updates the iterate of the medium parameter m_k along a search direction p_k to obtain $m_{k+1} = m_k + \alpha_k p_k$. To guarantee convergence, the search direction must be a descent direction, i.e., it must verify $\langle dJ(m_k)/dm, p_k\rangle < 0$. The step length α_k can be chosen to minimize the objective functional (1.9) along that search direction p_k. Solving that minimization problem exactly is too expensive for large-scale applications, as a single evaluation of the objective functional requires the solution of N_s PDEs. Instead, one can look for an approximate minimizer. The Wolfe conditions (1.23)-(1.24) safeguard how much approximation can be

introduced in the line search,

$$J(m_k + \alpha_k p_k) \leq J(m_k) + c_1 \alpha_k \frac{dJ}{dm}(m_k) p_k, \tag{1.23}$$

$$\frac{dJ}{dm}(m_k + \alpha_k p_k) p_k \geq c_2 \frac{dJ}{dm}(m_k) p_k, \tag{1.24}$$

with $0 < c_1 < c_2 < 1$. However, the curvature condition (1.24) still requires a new evaluation of the first derivative, i.e., the solution of an additional N_s PDEs. The computationally expensive curvature condition is therefore avoided by using a backtracking line search [32, 67]; the step length is computed by starting from an initial guess $\alpha_k^{(0)}$, then shortening until the sufficient decrease condition (1.23) is verified. When computing the search direction with a Newton-type method (see next paragraph), it is recommended to use $\alpha_k^{(0)} = 1$, as this is guaranteed to be a successful step length in a neighborhood of a minimum [67].

The high computational cost of PDE-constrained optimization makes the choice of the search direction critical to the success of the optimization. In the steepest descent method, one chooses the search direction to be the opposite of the first derivative, i.e., $p_k = -dJ/dm(m_k)$. This method is unfortunately extremely slow to converge, and is almost never a good choice. On the other hand, the convergence rate of Newton's method is quadratic in a neighborhood of a minimum. The Newton search direction is solution to the equation, $d^2J/dm^2(m_k)p_k = -dJ/dm(m_k)$. Therefore, the Newton search direction will only be a descent direction if the Hessian is positive definite; this is only guaranteed when m_k is in a neighborhood of a minimum. When the Hessian is indefinite, one solution is to replace the Hessian with a positive definite approximation, a common choice being the Gauss-Newton Hessian [70]. The Gauss-Newton Hessian is obtained by setting the adjoint variables $\{p_i\}_i$

to zero in the computation of the Hessian. Another option is to retain the full Hessian but solve the Newton system approximately, in a way that guarantees the computed solution to be a descent direction. For large-scale problems, assembling the Hessian requires too much memory. A solution is to solve the Newton system using the conjugate gradient (CG) method [81], which only requires the computation of Hessian-vector products (see section 1.3.1.2). The conjugate gradient method is a Krylov-subspace iterative method that solves linear systems of equations for symmetric positive definite matrices. Hence, the conjugate gradient method applied to a Newton system with an indefinite Hessian will eventually fail, as some intermediate quantities become negative. However, one can use the previous iterate of the solution, before the iteration failed, as the search direction; this choice is guaranteed to be a descent direction [67]. Often, it is recommended to add another criterion to terminate the CG iterations early, regardless of the definiteness of the Hessian. When far from a minimum, where Newton's method is not guaranteed to converge quadratically, the accurate computation of the Newton search direction is wasteful. Instead, it is more efficient to solve the Newton system with a coarse tolerance, and only apply a few iterations of the conjugate gradient method [27]. As the optimization converges, the tolerance should be gradually decreased to allow more and more accurate computation of the Newton search direction.

1.4 Full-waveform inversion

Full-waveform inversion describes a special case of inverse problem (1.1) for which observations $\{\mathbf{d}_i\}_i$ are produced by seismic waves. Although the oil & gas industry has traditionally relied on methods using simplified models

(see [9] for an extensive treatment of those), the approaching shortage of easily accessible fossil fuel energy forced oil & gas companies to turn to more precise imaging techniques. The promises of full-waveform inversion for enhanced accuracy are theoretically very high [85], making it an attractive technique for the future of seismic imaging. In regional and global-scale seismology, full-waveform inversion now represents the state-of-the-art for inferring earth properties from earthquake data [32, 35, 90]. Different types of mathematical models are used to represent the forward wave propagation in full-waveform inversion, and we discuss these alternatives in the first part of this section. In section 1.4.2, we study the inverse problem for the time-domain acoustic wave equation with absorbing boundary conditions, and use that example to illustrate some computational issues specific to full-waveform inversion.

1.4.1 The forward problem

We now introduce two mathematical models describing the propagation of seismic waves that appear in this dissertation, the time-domain and frequency-domain acoustic wave equations.

Time-domain acoustic wave equation The propagation of acoustic waves depends on the bulk modulus λ and the density ρ of the medium of propagation. Let us define the acoustic pressure, $u(\mathbf{x}, t) \coloneqq -\lambda(\mathbf{x})\nabla\cdot\mathbf{u}(\mathbf{x}, t)$, with $\mathbf{u}(\mathbf{x}, t)$ the displacement vector at the location $\mathbf{x}$ at time t. The time-domain acoustic wave equation [80], with absorbing boundary conditions [31] and ini-

tial conditions at rest is given by

$$\begin{cases} \dfrac{1}{\lambda}\ddot{u} - \nabla\cdot\left(\dfrac{1}{\rho}\nabla u\right) = f, & \text{in } \Omega\times(0,T), \\ u(\mathbf{x},0) = \dot{u}(\mathbf{x},0) = 0, & \text{in } \Omega, \\ \dfrac{1}{\rho}\nabla u\cdot\mathbf{n} = 0, & \text{on } \partial\Omega_n\times(0,T), \\ \dfrac{1}{\rho}\nabla u\cdot\mathbf{n} = -\dfrac{1}{\sqrt{\lambda\rho}}\dot{u}, & \text{on } \partial\Omega_a\times(0,T), \end{cases} \tag{1.25}$$

where f is a forcing term, $\dot{u}$ and $\ddot{u}$ are the first and second time derivatives of u, and the boundary of the domain is partitioned into $\partial\Omega = \partial\Omega_a \cup \partial\Omega_n$. The acoustic wave velocity of the medium is given by c, with the relation $\lambda = \rho c^2$. Equation (1.25) is a more general formulation of the acoustic wave equation than usually found in the literature. More commonly, the density ρ is assumed constant, which allows (1.25) to be re-written as

$$\frac{1}{c^2}\ddot{u} - \Delta u = \tilde{f}, \quad \text{in } \Omega\times(0,T). \tag{1.26}$$

Forcing term The forcing term $\tilde{f}$ in (1.26) corresponds to the divergence of the forcing term in the elastic wave equation $\mathbf{f}(\mathbf{x},t)$, i.e.,

$$\tilde{f} \coloneqq \nabla\cdot\left(\frac{1}{\rho}\mathbf{f}\right).$$

Depending on the application targeted, different levels of complexity are utilized to model the forcing term $\mathbf{f}$, ranging from the Ricker wavelet [9], popular in seismic exploration, to the more complex seismic moment tensor [2], a typical choice for global tomography. The Ricker wavelet, paired with a point source in space (e.g., explosive source [57]), leads to the forcing term

$$\mathbf{f}(\mathbf{x},t) = \left(1 - 2\pi^2 f_0^2 t^2\right) e^{-\pi^2 f_0^2 t^2} \delta(\mathbf{x}-\mathbf{x}_s)(1,1,1)^T.$$

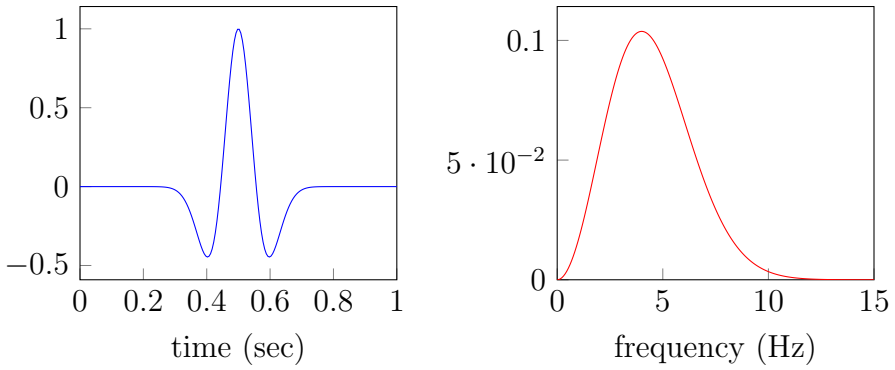


Figure 1.1: Time signal (left), and (non-negative) frequency content (right), of a Ricker wavelet centered at $f_0 = 4$Hz.

The Ricker wavelet has a compact frequency spectrum, peaking at the frequency f_0 (see figure 1.1). The seismic moment tensor, given by

$$\mathbf{f}(\mathbf{x}, t) = -M_{pq}(t)\frac{\partial}{\partial \mathbf{x}}\delta(\mathbf{x} - \mathbf{x}_s),$$

provides a way to model most types of fault mechanisms generating earthquakes. In this definition, $M_{pq}(t)$ models both the orientation of the earthquake and its loading pattern [2, 30], while $\mathbf{x}_s$ corresponds to the hypocenter of the earthquake.

Frequency-domain acoustic wave equation The frequency-domain acoustic wave equation is obtained by taking a Fourier transform, in time, of formulation (1.26) at a given frequency w,

$$-\frac{w^2}{c^2}\bar{u} - \Delta\bar{u} = \bar{f}, \quad \text{in } \Omega, \tag{1.27}$$

where $\bar{u}$ (resp. $\bar{f}$) is the Fourier transform in time of the quantity u (resp. $\tilde{f}$). Equation (1.27) is also known as the Helmholtz equation. The quantity w^2/c^2 is called the square of the wavenumber, and is often denoted by k, i.e., $k^2 = w^2/c^2$.

1.4.2 Inverse problem for the time-domain acoustic wave equation with absorbing boundary conditions

In this section, we assume that both the bulk modulus λ and the density ρ are unknown, and formulate an inverse problem governed by the acoustic wave equation (1.25). The choice of this inverse problem is motivated by two reasons: (1) we will use this inverse problem in chapter 3; and (2) the acoustic wave equation allows to discuss issues related to the time-discretization of the inverse problem. As both λ and ρ appear in (1.25) through their inverse, we introduce the parameters $\alpha \coloneqq 1/\lambda$ and $\beta \coloneqq 1/\rho$, and formulate the inverse problem in terms of α and β. As is typically the case in seismic inversion, we consider multiple experiments, characterized by their forcing term f_i and dataset $\mathbf{d}_i$. The acoustic wave inverse problem is then defined as

$$\min_{\alpha,\beta} \left\{ \frac{1}{2N_s} \sum_{i=1}^{N_s} \int_0^T |Bu_i(t) - \mathbf{d}_i(t)|^2 \, dt + \mathcal{R}(\alpha,\beta) \right\}, \tag{1.28}$$

where each u_i solves the forward problem (1.25) with forcing term f_i,

$$\begin{aligned} \alpha \ddot{u}_i - \nabla\cdot(\beta\nabla u_i) &= f_i, && \text{in } \Omega\times(0,T),\\ u_i(\mathbf{x},0) = \dot{u}_i(\mathbf{x},0) &= 0, && \text{in } \Omega,\\ \begin{cases} \beta\nabla u_i\cdot\mathbf{n} = 0, & \\ \beta\nabla u_i\cdot\mathbf{n} = -\sqrt{\alpha\beta}\dot{u}_i, & \end{cases} & && \begin{matrix} \text{on } \partial\Omega_n\times(0,T),\\ \text{on } \partial\Omega_a\times(0,T). \end{matrix} \end{aligned}$$

The question of the choice of the regularization term $\mathcal{R}(\alpha,\beta)$ in (1.28) is discussed in chapter 3. Let us introduce the function spaces,

$$\begin{aligned} \mathcal{V}_f &\coloneqq \left\{ u \in L^2(0,T;H^1(\Omega));\ \dot{u} \in L^2(0,T;L^2(\Omega));\ u(\mathbf{x},0) = \dot{u}(\mathbf{x},0) = 0 \right\},\\ \mathcal{V}_a &\coloneqq \left\{ v \in L^2(0,T;H^1(\Omega));\ \dot{v} \in L^2(0,T;L^2(\Omega));\ v(\mathbf{x},T) = \dot{v}(\mathbf{x},T) = 0 \right\}. \end{aligned}$$

We can then write the weak form of the forward problem as

$$\int_0^T \left((\alpha\ddot{u}_i, v) + (\beta\nabla u_i, \nabla v) + \int_{\partial\Omega_a} \sqrt{\alpha\,\beta}\,\dot{u}_i v \, ds - (f_i, v) \right) dt = 0, \tag{1.29}$$

for all admissible test functions $v \in \mathcal{V}_a$.

1.4.2.1 Derivatives of the inverse problem

Derivatives of the inverse problem (1.28) are computed through the Lagrangian approach introduced in section 1.3.1. Although the time dimension can be handled separately from the spatial dimensions (e.g., using a semi-discrete Lagrangian approach), we here follow a fully Lagrangian approach, and form the Lagrangian,

$$\mathscr{L}(\{u_i\}_i, \{v_i\}_i, \alpha, \beta) = \frac{1}{2N_s} \sum_{i=1}^{N_s} \int_0^T |Bu_i(t) - \mathbf{d}_i(t)|^2 \, dt + \mathcal{R}(\alpha)$$
$$+ \frac{1}{N_s} \sum_{i=1}^{N_s} \int_0^T \left((\alpha \ddot{u}_i, v_i) + (\beta \nabla u_i, \nabla v_i) + \int_{\partial\Omega_a} \sqrt{\alpha\,\beta}\, \dot{u}_i v_i \, ds - (f_i, v_i) \right) dt. \tag{1.30}$$

Gradient As detailed in section 1.3.1, the gradient of the inverse problem (1.28), in a direction $(\tilde{\alpha}, \tilde{\beta})$, is given by

$$\mathcal{R}'(\alpha, \beta)(\tilde{\alpha}, \tilde{\beta}) + \frac{1}{N_s} \sum_{i=1}^{N_s} \int_0^T \Big((\tilde{\alpha} \ddot{u}_i, v_i) + (\tilde{\beta} \nabla u_i, \nabla v_i)$$
$$+ \frac{1}{2} \int_{\partial\Omega_a} \left(\tilde{\alpha} \sqrt{\frac{\beta}{\alpha}} + \tilde{\beta} \sqrt{\frac{\alpha}{\beta}} \right) \dot{u}_i v_i \, ds \Big) \, dt, \tag{1.31}$$

where each u_i solves the state equation (1.29), and each v_i solves the adjoint equation,

$$\int_0^T \left((\alpha \tilde{u}, \ddot{v}_i) + (\beta \nabla \tilde{u}, \nabla v_i) - \int_{\partial\Omega_a} \sqrt{\alpha\,\beta}\, \tilde{u} \dot{v}_i \, ds + (B\tilde{u}, Bu_i - \mathbf{d}_i) \right) dt = 0, \tag{1.32}$$

for all admissible test functions $\tilde{u} \in \mathcal{V}_f$.

Hessian The action of the Hessian of the objective function (1.28) in a direction $(\hat{\alpha}, \hat{\beta})$ is given by

$$\mathcal{R}''(\alpha,\beta)(\hat{\alpha},\hat{\beta}) + \frac{1}{N_s}\sum_{i=1}^{N_s}\int_0^T \Big[(\tilde{\alpha}\ddot{\hat{u}}_i, v_i) + (\tilde{\alpha}\ddot{u}_i, \hat{v}_i) + (\tilde{\beta}\nabla\hat{u}_i, \nabla v_i) + (\tilde{\beta}\nabla u_i, \nabla\hat{v}_i)$$
$$+ \frac{1}{2}\int_{\partial\Omega_a}\left(\tilde{\alpha}\sqrt{\frac{\beta}{\alpha}} + \tilde{\beta}\sqrt{\frac{\alpha}{\beta}}\right)(\dot{\hat{u}}_i v_i + \dot{u}_i\hat{v}_i)\,ds$$
$$+ \frac{1}{4}\int_{\partial\Omega_a}\left(\frac{1}{\sqrt{\alpha\beta}}\tilde{\alpha}\hat{\beta} - \sqrt{\frac{\beta}{\alpha^3}}\tilde{\alpha}\hat{\alpha} + \frac{1}{\sqrt{\alpha\beta}}\tilde{\beta}\hat{\alpha} - \sqrt{\frac{\alpha}{\beta^3}}\tilde{\beta}\hat{\beta}\right)\dot{u}_i v_i\,ds\Big]\,dt, \quad (1.33)$$

for all admissible test functions $(\tilde{\alpha}, \tilde{\beta})$. Each variable $\hat{u}_i$ solves the incremental state equation,

$$\int_0^T\left((\alpha\ddot{\hat{u}}_i, \tilde{v}) + (\beta\nabla\hat{u}_i, \nabla\tilde{v}) + \int_{\partial\Omega_a}\sqrt{\alpha\,\beta}\,\dot{\hat{u}}_i\tilde{v}\,ds\right)dt =$$
$$-\int_0^T\left((\hat{a}\ddot{u}_i, \tilde{v}) + (\hat{\beta}\nabla u_i, \nabla\tilde{v}) + \frac{1}{2}\int_{\partial\Omega_a}\left(\hat{\alpha}\sqrt{\frac{\beta}{\alpha}} + \hat{\beta}\sqrt{\frac{\alpha}{\beta}}\right)\dot{u}_i\tilde{v}\,ds\right)dt,$$

and each variable $\hat{p}_i$ solves the incremental adjoint equation,

$$\int_0^T\left((\alpha\tilde{u}, \ddot{\hat{v}}_i) + (\beta\nabla\tilde{u}, \nabla\hat{v}_i) - \int_{\partial\Omega_a}\sqrt{\alpha\,\beta}\,\tilde{u}\dot{\hat{v}}_i\,ds\right)dt =$$
$$-\int_0^T\Big((\hat{\alpha}\tilde{u}, \ddot{v}_i) + (\hat{\beta}\nabla\tilde{u}, \nabla v_i) + (B\tilde{u}, B\hat{u}_i) - \frac{1}{2}\int_{\partial\Omega_a}\left(\hat{\alpha}\sqrt{\frac{\beta}{\alpha}} + \hat{\beta}\sqrt{\frac{\alpha}{\beta}}\right)\tilde{u}\dot{v}_i\,ds\Big)\,dt.$$

1.4.2.2 Discretization

A typical approach to discretize time-dependent PDEs is to handle the discretization in time and space separately. We are going to use continuous Galerkin finite elements in space, and a finite difference scheme in time. Starting with the spatial discretization, we introduce the finite-element space V_f (resp. V_a), as a finite dimensional approximation to the function

space $\mathcal{V}_f$ (resp. $\mathcal{V}_a$), with a basis $\{\phi_i\}_i$ (resp. $\{\varphi_i\}_i$). Let us now introduce the matrices, $\mathbf{M} = \left(\int_\Omega \alpha\phi_i\varphi_j\,dx\right)_{ij}$, $\mathbf{K} = \left(\int_\Omega \beta\nabla\phi_i\cdot\nabla\varphi_j\,dx\right)_{ij}$, and $\mathbf{D} = \left(\int_{\partial\Omega_a} \sqrt{\alpha\beta}\phi_i\varphi_j\,ds\right)_{ij}$. The semi-discrete formulation, in space, of the i^{th} forward PDE (1.29) at time t is given by

$$\mathbf{M}\ddot{\mathbf{u}}_i(t) + \mathbf{D}\dot{\mathbf{u}}_i(t) + \mathbf{K}\mathbf{u}_i(t) = \mathbf{f}_i(t),$$

where the dots denote time derivatives. We now deal with the time discretization. Let us introduce a time step Δt, and the discrete time variable $t_n = n\Delta t$ for $n = 0, \ldots, T/\Delta t$. The second order time derivative $\ddot{\mathbf{u}}_i$ is typically discretized using an explicit second-order centered scheme [53]. This type of time-discretization is particularly efficient when the mass matrix $\mathbf{M}$ is diagonal. In situations where the mass matrix is not diagonal, equally efficient solvers can be obtained by replacing the mass matrix with a diagonal approximation; this is also called matrix lumping. This type of approximation is discussed below. For the first-order time derivative $\dot{\mathbf{u}}_i$, on the other hand, we use a backward-in-time scheme. The motivation behind that choice is the more complicated structure of the derivatives for the damping term $\mathbf{D}\dot{\mathbf{u}}_i$; the combination of an explicit time-stepping with a lumped matrix would not, in that case, be beneficial, as it would require costly matrix assembly at each time step. The fully-discrete formulation of the i^{th} forward PDE at time t is given by

$$\mathbf{M}\frac{\mathbf{u}_i^{n+1} - 2\mathbf{u}_i^n + \mathbf{u}_i^{n-1}}{\Delta t^2} + \mathbf{D}\frac{\mathbf{u}_i^n - \mathbf{u}_i^{n-1}}{\Delta t} + \mathbf{K}\mathbf{u}_i^n = \mathbf{f}_i^n. \tag{1.34}$$

The semi-discrete formulation of the adjoint equation (1.32) is,

$$\mathbf{M}\ddot{\mathbf{v}}_i(t) - \mathbf{D}\dot{\mathbf{v}}_i(t) + \mathbf{K}\mathbf{v}_i(t) = -\mathbf{B}^T(\mathbf{B}\mathbf{u}_i(t) - \mathbf{d}_i(t)).$$

The adjoint equation is solved backward-in-time, i.e., starting at time $t = T$. To guarantee consistency of the discretized gradient with the discretized

cost functional, it is critical to select a time discretization for the adjoint equation that is the adjoint of the time discretization used for the forward equation. Centered difference schemes, like the one used for the acceleration term in (1.34), are self-adjoint, that is, whenever a centered difference scheme is used in the forward equation, the same centered difference scheme must be used in the adjoint equation. On the other hand, one-directional schemes are not self-adjoint. In the case of the backward-in-time scheme used for the time-derivative in (1.34), the adjoint time discretization is a forward-in-time scheme, which leads to the fully discrete adjoint equation

$$\mathbf{M}\frac{\mathbf{v}_i^{n+1} - 2\mathbf{v}_i^n + \mathbf{v}_i^{n-1}}{\Delta t^2} - \mathbf{D}\frac{\mathbf{v}_i^{n+1} - \mathbf{v}_i^n}{\Delta t} + \mathbf{K}\mathbf{v}_i^n = -\mathbf{B}^T(\mathbf{B}\mathbf{u}_i^n - \mathbf{d}_i^n).$$

We can verify this adjoint relation by deriving the discrete form of the adjoint equation through the discretize-then-optimize approach. To simplify the matter, we only look at a single time-step n, and a single source i. In the discrete Lagrangian, the only terms that involve the state variable $\mathbf{u}_i^n$ in the damping term are

$$(\mathbf{v}_i^n)^T\mathbf{D}\frac{\mathbf{u}_i^n - \mathbf{u}_i^{n-1}}{\Delta t} + (\mathbf{v}_i^{n+1})^T\mathbf{D}\frac{\mathbf{u}_i^{n+1} - \mathbf{u}_i^n}{\Delta t}.$$

Grouping these terms according to the state variable $\mathbf{u}_i^n$ gives the expression for the discrete damping term in the adjoint equation,

$$-\left(\frac{\mathbf{v}_i^{n+1} - \mathbf{v}_i^n}{\Delta t}\right)^T \mathbf{D}\mathbf{u}_i^n.$$

The solution to the forward PDE (1.34) is computed iteratively. Knowing the solutions at time steps $n-1$ and n, we compute the solution at time step $n+1$ with the formula,

$$\mathbf{u}_i^{n+1} = 2\mathbf{u}_i^n - \mathbf{u}_i^{n-1} + \Delta t^2\mathbf{M}^{-1}\left[\mathbf{f}_i^n - \mathbf{D}\frac{\mathbf{u}_i^n - \mathbf{u}_i^{n-1}}{\Delta t} - \mathbf{K}\mathbf{u}_i^n\right].$$

Each iteration requires the inversion of the matrix $\mathbf{M}$, a step that could be computationally intensive for large-scale systems. Instead, an efficient solution is to replace the original sparse matrix $\mathbf{M}$ with a diagonal approximation $\mathbf{M}_l$; this action is often called mass lumping. There exist several ways to diagonalize a mass matrix. A stable and accurate way to diagonalize a mass matrix is by keeping its diagonal, and rescaling it to maintain the same total mass as the full matrix [53]. Let us call $\mathrm{diag}(\mathbf{M})$ the diagonal of $\mathbf{M}$. We then define the lumped mass matrix as

$$\mathbf{M}_l \coloneqq \nu \,\mathrm{diag}(\mathbf{M}),$$

where $\nu \coloneqq \mathbf{1}^T \cdot \mathbf{M} \cdot \mathbf{1}/\mathrm{tr}(\mathbf{M})$, and $\mathbf{1}$ is a vector of one's. The use of a lumped mass matrix in the forward, adjoint, incremental forward, and incremental adjoint equations require to modify the gradient (1.31) and Hessian-vector product (1.33) expressions to maintain consistency. Indeed, with a full mass matrix, the term $(\tilde{\alpha}\ddot{u}_i(t), v_i(t))$ in (1.31) (resp. $(\tilde{\alpha}\ddot{\hat{u}}_i(t), v_i(t))$ in (1.33)) would become, after discretization, $(\mathbf{v}_i^n)^T \mathbf{M}\ddot{\mathbf{u}}_i^n$ (resp. $(\mathbf{v}_i^n)^T \mathbf{M}\ddot{\hat{\mathbf{u}}}_i^n$). With a lumped mass matrix, for general functions $u = \sum_j u_j \varphi_j$ and $v = \sum_i v_i \phi_i$, we obtain

$$\mathbf{v}^T \mathbf{M}_l \mathbf{u} = \nu \sum_i u_i v_i \int_\Omega \tilde{\alpha} \phi_i \varphi_i \, dx.$$

Let us define the matrix $(\mathbf{M}_l')_{ij} = \nu \int_\Omega \psi_i \phi_j \varphi_j \, dx$, and the vector $\mathbf{w}_i = u_i v_i$. We can then write,

$$\frac{\partial}{\partial \alpha} \left(\mathbf{v}^T \mathbf{M}_l \mathbf{u} \right) = \mathbf{M}_l' \mathbf{w}.$$

Provided one is allowed to store the matrix $\mathbf{M}_l'$, the gradient and Hessian-vector products of the lumped mass matrix can be computed efficiently.

1.5 Scope of research

With the increase in compute power and the advent of the big data era, inverse problems have grown more complex, attempting to extract more information and to use more data. While this evolution manifests itself in multiple forms, we focus in this dissertation on three specific aspects: multi-source, multi-parameter, and multi-physics inverse problems.

1.5.1 Multi-source inverse problems

As discussed in section 1.3, the numerical computation of the gradient of an inverse problem, or the application of the Hessian along a direction, scales linearly with the number of sources N_s. Therefore both gradient-based methods and Newton-type methods become impractical for large-scale applications relying on a large number of sources. This is, for instance, the case of full-waveform inversion, where 3D large-scale applications routinely use on the order of 1,000 experiments. There have been some recent breakthroughs to address this computational bottleneck using the concept of random source encoding, sometimes also referred to as simultaneous random sources [59, 73].

In [43] the authors give a mathematical justification of the idea of random source encoding for a discrete problem and we apply their argument, here, to the general inverse problem (1.1). We gather all $\mathcal{F}_i(m)$ (resp. $\mathbf{d}_i$) into the columns of a matrix $\mathbf{F}(m)$ (resp. $\mathbf{D}^e$) and call the data misfit matrix $\mathbf{S}(m) \coloneqq \mathbf{F}(m) - \mathbf{D}^e$. Ignoring the regularization term for now, the inverse problem can be written as, $\min_{m\in\mathcal{V}} \left\{ \|\mathbf{S}(m)\|_F^2 \right\}$, where $\|\cdot\|_F$ is the Frobenius norm [81]. Note that $\|\mathbf{S}(m)\|_F^2 = \text{trace}(\mathbf{S}(m)^T\mathbf{S}(m))$, which can be approximated efficiently using randomized trace estimators [8, 54]. Indeed, for random vectors $\mathbf{z}$ with mean zero and identity covariance matrix,

one finds that, $\text{trace}(\mathbf{S}(m)^T\mathbf{S}(m)) = \mathbb{E}_{\mathbf{z}}\big(\|\mathbf{S}(m)\mathbf{z}\|_2^2\big)$. Typical choices of distribution for $\mathbf{z}$ include the Rademacher distribution, where samples take values ± 1 with probability $1/2$, and the standard normal distribution $\mathcal{N}(0, \mathbf{I}_{N_s})$. Among other possible choices we mention the discrete distribution that takes values $\pm\sqrt{3}$ with probability $1/6$ and 0 otherwise, or the uniform spherical distribution on a sphere of radius $\sqrt{N_s}$ that we denote $\mathcal{U}(\sqrt{N_s})$; the fact that $\mathcal{U}(\sqrt{N_s})$ has identity covariance matrix can be shown using results from [5], along with the observation that $\tilde{z} \sim \mathcal{U}(\sqrt{N_s})$ iff $\tilde{z} = \sqrt{N_s}(\mathbf{z}/|\mathbf{z}|)$ with $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I}_{N_s})$. We now write the data-misfit term as an expectation, i.e., $\|\mathbf{F}(m) - \mathbf{D}^e\|_F^2 = \mathbb{E}_{\mathbf{z}}(\|(\mathbf{F}(m) - \mathbf{D}^e)\mathbf{z}\|^2)$, leading to the stochastic optimization problem

$$\min_{m \in \mathcal{V}} \Big\{ \mathbb{E}_{\mathbf{z}}\big(\|(\mathbf{F}(m) - \mathbf{D}^e)\mathbf{z}\|^2\big) \Big\}.$$

There exist two main techniques to solve these types of problems [76]. Using stochastic average approximation (SAA), one approximates the cost functional with a Monte-Carlo-type approach before solving a deterministic optimization problem, i.e., for fixed samples $\mathbf{z}_i$ ones solves

$$\mathbb{E}_{\mathbf{z}}\big(\|(\mathbf{F}(m) - \mathbf{D}^e)\mathbf{z}\|^2\big) \approx \frac{1}{M} \sum_{i=1}^{M} \|(\mathbf{F}(m) - \mathbf{D}^e)\mathbf{z}_i\|^2.$$

In an alternative approach called stochastic approximation (SA), one re-samples the random vector $\mathbf{z}$ at each step of the iteration.

The method of random source encoding, stochastic in essence, suffers from a few limitations. The key idea of the random source encoding approach is the conversion of the deterministic optimization (1.1) into a stochastic optimization problem. The expectation to be minimized is then approximated using a Monte-Carlo technique. To reduce the computational cost of solving the inverse problem, one would like to choose the number of samples used in

this Monte-Carlo approximation small. A small number of samples translates into a large variance for the Monte-Carlo estimator of the expectation. In practice, this manifests itself in large differences in the reconstructions obtained with different samples of encoding weights. An approach to remedy that difficulty is to select the weights deterministically [48, 79]. In particular, in [79], the author considers to select the weights that generate the greatest improvement from the current reconstruction, but the results are inconclusive. In [48], the authors choose the weights that minimize the expected medium misfit in the case of a discrete linear inverse problem. In chapter 2, we propose a new way to select the weights deterministically, that is related to [48].

1.5.2 Multi-parameter inverse problems

We define a joint inverse problem as the simultaneous inference of n distinct parameter fields. In this dissertation, we consider two types of joint inverse problems. In the first case, we have at our disposal a single type of observations $\mathbf{d}$, generated from a single physical phenomenon which depends on multiple parameter fields $\{m_i\}_{i=1}^n$. The parameter-to-observable map is defined as $\mathcal{F}(m_1, \ldots, m_n) = B\big[\mathcal{A}(m_1, \ldots, m_n)\big]^{-1} f$. A prototypical formulation for that type of joint inverse problem is given by

$$\min_{\{m_i\}_i} \left\{ \frac{1}{2} |\mathcal{F}(m_1, \ldots, m_n) - \mathbf{d}|^2 + \mathcal{R}(m_1, \ldots, m_n) \right\}. \tag{1.35}$$

The role played by the term $\mathcal{R}$ in (1.35) is discussed in section 1.5.2.1. One example of joint inverse problem (1.35) is the inversion for both the primary and secondary wave speeds inside the Earth given recordings of the ground acceleration at the surface [32, 60, 63]. In the second type of joint inverse problem we are interested in, we utilize n datasets, generated from physical phenomena that depend on different parameter fields. Without loss of

generality, let us assume that each physical phenomenon depends on a single parameter field m_i, and let us call $\mathbf{d}_i$ the dataset corresponding to the physical phenomenon depending on parameter m_i. The parameter-to-observable maps are defined by $\mathcal{F}_i(m_i) = B_i\big[\mathcal{A}_i(m_i)\big]^{-1} f_i$. The formulation for this type of joint inverse problem is given by

$$\min_{\{m_i\}_i} \left\{ \frac{1}{2} \sum_{i=1}^{n} |\mathcal{F}_i(m_i) - \mathbf{d}_i|^2 + \mathcal{R}(m_1, \ldots, m_n) \right\}. \tag{1.36}$$

Among the different physical phenomena that can be combined, we mention electromagnetic and seismic waves [1, 75], DC resistivity and seismic wave [37], or current resistivity and groundwater flow [77].

1.5.2.1 The joint regularization term

The role of the joint regularization term $\mathcal{R}(m_1, \ldots, m_n)$ in (1.35) and (1.36) is twofold. It plays the role of a regularization term for each parameters $\{m_i\}_i$, and it introduces coupling between the parameters, if desired. In the case of a joint inverse problem (1.36), coupling may be desirable when the parameter fields are known in advance to share a similar structure. Or in the case of a joint inverse problem (1.35), this coupling may help improve the quality of the reconstruction. For instance, in the case of full-waveform inversion, despite the wave equation coupling both wave speeds, it is notoriously difficult to obtain good quality reconstructions for both parameters without providing additional coupling to the inverse problem [32].

Both regularization and coupling in $\mathcal{R}(m_1, \ldots, m_n)$ can be handled separately, or in a unified fashion. To separate regularization and coupling, we

can decompose the joint regularization term $\mathcal{R}(m_1, \ldots, m_n)$ as

$$\mathcal{R}(m_1, \ldots, m_n) = \sum_{i=1}^{n} \mathcal{R}_i(m_i) + \hat{\mathcal{R}}(m_1, \ldots, m_n),$$

where the terms $\mathcal{R}_i$ are regularization terms for each medium parameter m_i, and are typically chosen to be Tikhonov regularizations or total variation regularizations. The structural similarity term $\hat{\mathcal{R}}(m_1, \ldots, m_n)$ defines the coupling between all parameters $\{m_i\}_i$. We now list a few possible choices for the structural similarity term $\hat{\mathcal{R}}$. In [37], the authors introduced the cross-gradient term for two parameters m_1 and m_2,

$$\hat{\mathcal{R}}_{\text{cg}}(m_1, m_2) = \frac{1}{2} \int_\Omega |\nabla m_1 \times \nabla m_2|^2 \, dx.$$

It is today the most popular choice in geophysics [1, 37, 75, 77]. Instead of dealing with the gradient of each parameter, one can instead use normalized gradients, and obtain the normalized cross-gradient term

$$\mathcal{R}_{\text{ncg}}(m_1, m_2) = \int_\Omega \left| \frac{\nabla m_1}{|\nabla m_1|} \times \frac{\nabla m_2}{|\nabla m_2|} \right|^2 dx.$$

The normalized cross-gradient was first introduced in the context of image registration [47]. The main idea behind both of these structural similarity terms is to force the level sets of both inversion parameters, m_1 and m_2, to align. Alternatively, when an empirical constitutive relation between all parameters is known, one could use it in place of the structural similarity term $\hat{\mathcal{R}}$ [1, 44]. This approach is however not recommended in practice as these relations are typically uncertain and the resulting optimization problem is harder to solve [38, 44].

On the other hand, one can apply both regularization and coupling through a single functional by using, for instance, the vectorial total varia-

tion (VTV),

$$\mathcal{R}_{\text{VTV}}(m_1, \ldots, m_n) = \int_\Omega \sqrt{\sum_{i=1}^{n} |\nabla m_i|^2}\, dx.$$

The VTV functional was first introduced in the context of multi-channel imaging [12, 14], and later applied to PDE-constrained joint inverse problems [44]. In [58], the authors introduced a vector-valued equivalent of the total generalized variation, using the nuclear norm to enforce gradient alignment. Building on that idea, we introduce in chapter 3 a novel joint regularization term for PDE-constrained joint inverse problems, based on the nuclear norm. In chapter 3, we also analyze in details the four joint regularizations introduced above, and conduct a comprehensive numerical comparison of these functionals over three joint inverse problems covering both types of joint inverse problems (1.35) and (1.36).

1.5.2.2 Efficient solver for the solution of joint inverse problems regularized with VTV

Because the evaluation of the objective function and the computation of the gradient each require the solution of the forward PDE N_s times, an efficient solver must use curvature information, i.e., be of Newton type. However, the straightforward use of Newton linearization for TV regularization leads to extremely poor performance [86]; the number of PDE solves required to converge increases dramatically when the hyperparameter ε in (1.6) decreases, corresponding to reconstructions with increasingly sharper edges. Because of the strong connection between TV and VTV, similar numerical difficulties are expected when using Newton method to solve a joint inverse problem regularized with VTV. Building on the work of [19, 50], we introduce, in

chapter 4, a scalable primal-dual Newton method for joint inverse problems regularized with VTV.

1.5.3 Can a multi-physics inverse problem provide low-frequency information for full-waveform inversion?

A major impediment of full-waveform inversion is the existence of multiple local minima in the objective function when using high frequency sources; this phenomenon is also known as cycle skipping. This happens, for instance, by shifting along the time component of the recorded seismogram; in a layered medium this corresponds to applying constant perturbations to each layer. More generally the least-squares data misfit part of the objective functional (1.28) is oscillatory along smooth directions of the Lamé parameters [32].

The traditional remedy is a continuation scheme over the frequency of the source term, also called multilevel waveform inversion [16]. The idea of frequency continuation is to solve a sequence of inverse problems with sources of increasing frequencies, using the reconstruction from the lower-frequency as initial guess for the higher frequency problem. The restriction in frequency is achieved by filtering the frequency content of data and sources. Because the spatial grid size is dictated by the frequency content of the problem, one can adjust the grid size to the frequency level. Although this affects the quality of the reconstruction, this allows important savings in compute time. The key observation to motivate that approach is that the objective function is less oscillatory for low-frequency waves and therefore a good initial guess has a better chance of lying within the basin of attraction of the global minimum.

A major drawback of the frequency continuation approach is that low-

frequency data is rarely available in practice. One solution is to replace the missing data with low-frequency information coming from a different physics, or a different type of inversion, e.g., first-time arrival [89]. In chapter 3, we investigate how a Poisson inverse problem can be used to provide the missing low-frequency information for full-waveform inversion, provided the truth parameter fields for the Poisson inverse problem and full-waveform inversion have similar structures. We connect both inverse problems through the formulation of a multi-physics inverse problem.

Chapter 2

A-optimal encoding weights for nonlinear inverse problems, with application to the Helmholtz inverse problem

Drawing from recent developments in optimal experimental design for high- or infinite-dimensional inverse problems [3, 4, 45, 46], and following a Bayesian view of inverse problems, we develop a method for the computation of encoding weights that lead to a parameter reconstruction with the least uncertainty—as measured by the average of the posterior variance. We refer to these (deterministic) weights as *A-optimal encoding weights*, a nomenclature motivated by the use of the A-optimal experimental design criterion from optimal experimental design theory [83]. The method we propose extends the work in [48] by addressing inverse problems with nonlinear parameter-to-observable maps, and allows for infinite-dimensional parameters. The infinite-dimensional formulation has two main advantages: (a) the use of weak forms facilitates the derivation of adjoint-based expressions for the gradient of the objective function to compute the A-optimal encoding weights; (b) it allows us to follow the optimize-then-discretize approach, which, along with devoting special attention to the discretization of the formulation and the choice of the numerical methods employed, helps control the computational cost independently of the parameter discretization. We elaborate our method for the Helmholtz inverse problem and derive the adjoint-based gradient of the optimization problem for

finding the A-optimal encoding weights. We also analyze the computational cost—in terms of Helmholtz PDE solves—of objective and gradient evaluation for this optimization problem. For this Helmholtz problem, we present an extensive numerical study and discuss the potential and pitfalls of our approach. The results from this chapter were published in [22][1].

The rest of this chapter is organized as follows. In section 2.1, we summarize elements of Bayesian inverse problems and introduce approximations to the posterior covariance in function space. The framework for the A-optimal encoding weights is presented in section 2.2. In section 2.3, we elaborate our formulation for the Helmholtz inverse problem. We derive adjoint-based expressions for the gradient of the A-optimal objective function, and analyze computational cost of evaluating the objective function and its gradient. Numerical results are presented in section 2.4, and we provide some concluding remarks in section 2.5.

2.1 Bayesian formulation of the inverse problem with encoded sources

This section contains a brief presentation of the Bayesian formulation of inverse problems with infinite-dimensional inversion parameters; for details we refer the reader to [25, 78] for theory and to [15] for the numerical approximation. In the Bayesian framework, the unknown parameter function m is modeled as a random field. Starting from a prior distribution law for m, we

[1] [22] was co-authored by the author, Alen Alexanderian, Georg Stadler, and Omar Ghattas. The author implemented the numerical methods presented in that work; the presentation of those methods, and the design of the numerical experiments testing them, was a collaborative effort between the author, Prof. Alexanderian, and Prof. Stadler; Prof. Ghattas aided in the editing of the work, provided guidance, and advised.

use observation data to obtain an improved description of the law of m. This updated distribution law of m is called the posterior measure. The prior measure, which we denote by μ_0, can be understood as a probabilistic model for our prior beliefs about the parameter field m. The posterior measure, which we denote by μ_{post}, is the distribution law of m, conditioned on observation data. A key ingredient of a Bayesian inverse problem is the data likelihood, $\pi_{\text{like}}(\mathbf{d}|m)$, which describes the conditional distribution of the data given the parameter field m; this is where the parameter-to-observable map enters the Bayesian inverse problem.

Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with piecewise smooth boundary and $(S, \Sigma, \mathbb{P})$ a probability space. We consider an inference parameter $m = m(x, \omega)$, with $(x, \omega) \in \Omega \times S$, such that for any $\omega \in S$, $m(\cdot, \omega) \in \mathcal{V}$ where $\mathcal{V}$ is an infinite-dimensional Hilbert space. Considering the law of m as a probability measure on $(\mathcal{V}, \mathfrak{B}(\mathcal{V}))$, the infinite-dimensional Bayes' theorem relates the Radon-Nikodym derivative of μ_{post} with respect to μ_0 with the data likelihood $\pi_{\text{like}}(\mathbf{d}|m)$:

$$\frac{d\mu_{\text{post}}}{d\mu_0} \propto \pi_{\text{like}}(\mathbf{d}|m). \tag{2.1}$$

The use of non-Gaussian priors in infinite-dimensional Bayesian inverse problems represents a new, interesting area of research (see for instance [25, 52]). However, since the Bayesian inverse problem, in the formulation we introduce in section 2.2, only represents the inner problem, the additional complications created by the use of non-Gaussian priors are not justified. We instead rely on Gaussian priors for the Bayesian inverse problem; i.e., $\mu_0 = \mathcal{N}(m_0, \mathcal{C}_0)$ is a Gaussian measure on $\mathcal{V}$. In that case, we require $\mathcal{C}_0$ to be symmetric, positive and trace-class [78]. A common choice for $\mathcal{C}_0$ (in two and three space dimensions) is the squared inverse of a Laplacian-like operator $\mathcal{K}$, i.e., $\mathcal{C}_0 = \mathcal{K}^{-2}$.

We also assume that the noise in the data is additive, and independent and identically distributed (over the different experiments); the distribution of each noise vector is normal with mean zero and covariance matrix $\mathbf{\Gamma}_{\text{noise}}$. That is, $\mathbf{d}_i|m \sim \mathcal{N}\big(\mathcal{F}_i(m), \mathbf{\Gamma}_{\text{noise}}\big)$, for any $i \in \{1, \dots, N_s\}$. Consequently, each encoded observation $\mathbf{d}(\mathbf{w}^i)$ will be normally distributed with mean zero and covariance matrix $\mathbf{\Gamma}_{\text{noise,i}} \coloneqq (\sum_{j=1}^{N_s}(w_j^i)^2)\mathbf{\Gamma}_{\text{noise}}$, i.e., $\mathbf{d}(\mathbf{w}^i)|m \sim \mathcal{N}\big(\mathcal{F}(\mathbf{w}^i; m), \mathbf{\Gamma}_{\text{noise,i}}\big)$, for $i \in \{1, \dots, N_w\}$. Therefore, the likelihood function has the form

$$\pi_{\text{like}}(\mathbf{d}(\mathbf{w})|m) \propto \exp\left(-\frac{1}{2N_w}\sum_{i=1}^{N_w} \|\mathcal{F}(\mathbf{w}^i; m) - \mathbf{d}(\mathbf{w}^i)\|^2_{\mathbf{\Gamma}^{-1}_{\text{noise,i}}}\right).$$

2.1.1 MAP point

In finite dimensions, the MAP point is the parameter m_{MAP} that maximizes the posterior probability density function. Although this definition does not extend directly to the infinite-dimensional case, a MAP point can still be defined as a minimizer of a regularized data-misfit cost functional over an appropriate Hilbert subspace of the parameter space [78]. Let us define the Cameron-Martin space $\mathbb{E} = Im(\mathcal{C}_0^{1/2})$, endowed with the inner-product

$$\langle x, y\rangle_{\mathbb{E}} \coloneqq \langle \mathcal{C}_0^{-1/2}x, \mathcal{C}_0^{-1/2}y\rangle = \langle \mathcal{K}x, \mathcal{K}y\rangle, \quad \forall x, y \in \mathbb{E}. \tag{2.2}$$

Then the MAP point is defined as

$$m_{\text{MAP}}(\mathbf{w}) = \arg\min_{m\in\mathbb{E}} \{\mathcal{J}(\mathbf{w}; m)\}, \tag{2.3}$$

where, for the inverse problems considered in the present work, the functional $\mathcal{J}(\mathbf{w}; \cdot) : \mathbb{E} \to \mathbb{R}$ is defined as

$$\mathcal{J}(\mathbf{w}; m) \coloneqq \frac{1}{2N_w}\sum_{i=1}^{N_w} \left\|\mathcal{F}(\mathbf{w}^i; m) - \mathbf{d}(\mathbf{w}^i)\right\|^2_{\mathbf{\Gamma}^{-1}_{\text{noise,i}}} + \frac{1}{2}\|m - m_0\|^2_{\mathbb{E}}. \tag{2.4}$$

Here, the function $m_0 \in \mathbb{E}$ is the mean of the prior measure.

2.1.2 Approximation to the posterior covariance

In general, there are no closed-form expressions for moments of the posterior measure. Thus, one usually relies on sampling-based methods to explore the posterior. For inverse problems governed by PDEs and problems with high-dimensional parameters (as, for instance, arising upon discretization of an infinite-dimensional parameter field), sampling of the posterior can quickly become infeasible since every evaluation of the likelihood requires a PDE solve. We thus rely on approximations of the posterior, namely Gaussian approximations about the MAP estimate. After finding the MAP point, we consider two commonly used approximations of the posterior measure by a Gaussian measure $\mathcal{N}(m_{\text{MAP}}, \mathcal{C}_{\text{post}})$, as discussed next [15, 65].

Gauss–Newton approximation Assuming the parameter to observable map $\mathcal{F}(\mathbf{w}^i; \cdot\,)$ is Fréchet differentiable at the MAP point, one strategy to approximate the posterior is to linearize around the MAP point, i.e.,

$$\mathcal{F}(\mathbf{w}^i; m) \approx \mathcal{F}(\mathbf{w}^i; m_{\text{MAP}}) + \mathbf{J}_{\mathbf{w}^i}(m - m_{\text{MAP}}),$$

with $\mathbf{J}_{\mathbf{w}^i} : \mathcal{V} \to \mathbb{R}$ the Fréchet derivative of the parameter-to-observable map $\mathcal{F}(\mathbf{w}^i; \cdot\,)$ evaluated at the MAP point (2.3). Calling $(\mathbf{J}_{\mathbf{w}^i})^*$ the adjoint of $\mathbf{J}_{\mathbf{w}^i}$, the covariance operator of the resulting Gaussian approximation of the posterior is given by

$$\mathcal{C}^G_{\text{post}} = \left(\frac{1}{N_w} \sum_{i=1}^{N_w} (\mathbf{J}_{\mathbf{w}^i})^* \mathbf{\Gamma}^{-1}_{\text{noise,i}} \mathbf{J}_{\mathbf{w}^i} + \mathcal{C}_0^{-1} \right)^{-1}. \tag{2.5}$$

Note that the operator that appears inside the brackets in (2.5) is the so called Gauss–Newton Hessian of the functional (2.4) evaluated at the MAP point,

$$\mathcal{H}_{\text{GN}}(m_{\text{MAP}}) := \frac{1}{N_w} \sum_{i=1}^{N_w} (\mathbf{J}_{\mathbf{w}^i})^* \mathbf{\Gamma}^{-1}_{\text{noise,i}} \mathbf{J}_{\mathbf{w}^i} + \mathcal{C}_0^{-1}.$$

Laplace approximation Assuming $\mathcal{J}(\mathbf{w};\cdot)$, in (2.4), is at least twice Fréchet differentiable at the MAP point, a second approach called Laplace approximation consists of using the second derivative of $\mathcal{J}(\mathbf{w};\cdot)$, i.e., the Hessian, at the MAP point as an approximation to the posterior covariance

$$\mathcal{C}_{\text{post}}^{L} = \left(\mathcal{J}''(\mathbf{w}; m_{\text{MAP}})\right)^{-1} = \mathcal{H}^{-1}(m_{\text{MAP}}), \tag{2.6}$$

where the derivative in $\mathcal{J}''$ is taken in terms of the parameter field m. Note that the Laplace approximation can be related, in finite dimensions, to a quadratic local approximation of $\mathcal{J}(\mathbf{w};\cdot)$ around the MAP point.

2.2 A-optimal approach to source encoding

Combining the results from section 2.1 with elements from optimal experimental design, we propose a rigorous method to compute A-optimal encoding weights. In the Bayesian framework, the posterior covariance quantifies the uncertainty in the reconstruction. Since the posterior covariance depends on the weights (see section 2.2.1), we can select the weights that lead to a reconstruction with the least uncertainty. In the field of optimal experimental design, there are various design criteria that measure the statistical quality of the reconstructed parameter field [71]. In the present work, we rely on the A-optimal design criterion [7, 71], which aims to minimize the trace of the posterior covariance, or equivalently, to minimize the average posterior variance. That is, we compute the weights with the smallest trace of the posterior covariance $\Phi(\mathbf{w}) = \text{tr}(\mathcal{C}_{\text{post}})$, with $\mathcal{C}_{\text{post}}$ given by $\mathcal{C}_{\text{post}}^{G}$ (2.5) or $\mathcal{C}_{\text{post}}^{L}$ (2.6).

An alternate view of the A-optimal design criterion is that of minimizing the expected Bayes risk of the MAP estimator, which coincides with the trace of the posterior covariance for a linear inverse problem [3, 17, 45]. This

interpretation of the A-optimal criterion can be stated as the average mean squared error between the MAP estimator (i.e., the parameter reconstruction) and the true parameter (e.g., see [3]). While this interpretation of A-optimality is restricted to linear inverse problems, it provides another motivation for our choice of the design criterion. In our numerical results, we explore this relation between minimizing the trace of the posterior covariance and the mean squared distance between the MAP point and the true parameter and observe that minimizing the trace of the posterior covariance correlates with smaller errors for the parameter reconstruction.

2.2.1 Dependence of the operators $\mathcal{C}^G_{\text{post}}$ and $\mathcal{C}^L_{\text{post}}$ on w

The dependence of the operators $\mathcal{C}^G_{\text{post}}$ (2.5) and $\mathcal{C}^L_{\text{post}}$(2.6) on the weights is twofold. First these operators depend on the encoded parameter to observable maps that depend explicitly on the weights, $\mathcal{F}(\mathbf{w}^i; m) = \sum_{j=1}^{N_s} w_j^i \mathcal{F}_j(m)$. Moreover, the posterior covariance operators also depend on the weights through the MAP point (2.3), which depends on the weights as illustrated by (2.3) and (2.4).

The dependence of the covariance operator $\mathcal{C}^G_{\text{post}}$ on $\mathbf{w}$ is straightforward to see. In particular, using the chain-rule on the forward problem $\mathcal{A}(m)u_i = f(\mathbf{w}^i)$, the Fréchet derivative of the parameter-to-observable at the MAP point is given by

$$\mathbf{J}_{\mathbf{w}^i} = -B\mathcal{A}(m_{\text{MAP}}(\mathbf{w}))^{-1} \frac{\partial \mathcal{A}(m)u_i}{\partial m}\bigg|_{m=m_{\text{MAP}}(\mathbf{w})}. \tag{2.7}$$

Given N_w encoding weights $\mathbf{w} = (\mathbf{w}^1, \ldots, \mathbf{w}^{N_w})$ where $\mathbf{w}^i \in \mathbb{R}^{N_s}$, we emphasize the dependence of the posterior covariance on the weights by writing $\mathcal{C}^G_{\text{post}} = \mathcal{C}^G_{\text{post}}(\mathbf{w})$. The structure of the covariance operator $\mathcal{C}^L_{\text{post}}$ is more com-

plicated. We detail the dependence of $\mathcal{C}_{\text{post}}^L$ on $\mathbf{w}$ for the application problem considered in the present chapter in section 2.3. Note that in the case of a linear parameter-to-observable map, both posterior covariances (2.5) and (2.6) are equal.

In the present formulation, $\text{tr}\big(\mathcal{C}_{\text{post}}(\mathbf{w})\big)$ scales with the weights. For instance, applying a constant multiplicative factor $\lambda > 1$ to all weights would reduce the influence of the prior in the computation of the MAP point (2.3) for once. It would also inflate the norm of the state variable u_i by that factor λ, which would then increase the size of the derivative (2.7). This would in turn artificially reduce the trace of the posterior covariance (2.5). A solution is to restrict the codomain of each encoding weight to a sphere of radius r in $\mathbb{R}^{N_s}$. We denote the corresponding space, for the weights $\mathbf{w}$, by $\mathbf{S}_r$, i.e., $\mathbf{S}_r \coloneqq \big\{\mathbf{w} = (\mathbf{w}^1, \dots, \mathbf{w}^{N_w}) \in \mathbb{R}^{N_w N_s};\, |\mathbf{w}^i| = r,\, \forall i\big\}$. As discussed in section 1.5.1, the theory of randomized trace estimation dictates the use of $r = \sqrt{N_s}$. However this value is arbitrary and can be compensated by an equivalent re-scaling of the regularization parameter. Therefore for simplicity we use the value $r = 1$ along with the notation $\mathbf{S} \coloneqq \mathbf{S}_1$. Another implication of that choice, $|\mathbf{w}^i| = 1$, is that the covariance matrices for the encoded noise vectors, introduced in section 2.1, simplify to $\mathbf{\Gamma}_{\text{noise,i}} = \mathbf{\Gamma}_{\text{noise}}$, for $i \in \{1, \dots, N_w\}$.

2.2.2 A-optimal encoding weights

We propose to compute the A-optimal encoding weights as the solution to the constrained minimization problem

$$\min_{\mathbf{w} \in \mathbf{S}} \Phi(\mathbf{w}) \coloneqq \text{tr}\big(\mathcal{C}_{\text{post}}(\mathbf{w})\big). \tag{2.8}$$

Since there are no closed-form expressions for moments of the posterior measure, we replace the exact posterior covariance in (2.8) with one of the two

approximations introduced in section 2.1.2. The Gauss–Newton formulation of the A-optimal encoding weights,

$$\Phi_{\mathrm{GN}}(\mathbf{w}) = \mathrm{tr}(\mathcal{H}_{\mathrm{GN}}^{-1}(\mathbf{w}; m_{\mathrm{MAP}}(\mathbf{w}))), \tag{2.9}$$

is based on the posterior covariance approximation (2.5), and the Laplace formulation,

$$\Phi_{\mathrm{L}}(\mathbf{w}) = \mathrm{tr}(\mathcal{H}^{-1}(\mathbf{w}; m_{\mathrm{MAP}}(\mathbf{w}))), \tag{2.10}$$

is based on the posterior covariance (2.6). Note that both formulations (2.9) and (2.10) require the computation of the MAP point which is computationally expensive for large-scale problems. To avoid the cost associated with the computation of the MAP point, an additional simplification of (2.9) can be achieved by evaluating the posterior covariance (2.5) at a reference parameter field m_0, which leads to the following (simplified) objective function,

$$\Phi_0(\mathbf{w}) = \mathrm{tr}(\mathcal{H}_{\mathrm{GN}}^{-1}(\mathbf{w}; m_0)). \tag{2.11}$$

A-optimal encoding weights formulation for large-scale applications Formulation (2.8) is a nonlinear optimization problem that requires the use of iterative methods. These methods involve repeated evaluations of the trace of the posterior covariance. Following discretization, the posterior covariance is a high-dimensional operator that is defined implicitly, i.e., through its applications to vectors. The exact computation of the trace of such operators, and their derivatives with respect to encoding weights, is computationally intractable. For this reason, we propose an approximate formulation using a randomized trace estimator (see [8, 54] for the theory, and [3, 48] for examples of applications). Following the formulation in [4], we introduce the Gaussian measure $\mu_\delta = \mathcal{N}(0, \mathcal{C}_\delta)$ where $\mathcal{C}_\delta := (I - \delta\Delta)^{-2}$. Here Δ denotes the Laplacian

operator with homogeneous Neumann boundary conditions and $\delta > 0$ a sufficiently small real number. Then for any positive, self-adjoint and trace-class operator $\mathcal{T}$, we may use an estimator of the form,

$$\mathrm{tr}(\mathcal{T}) \approx \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \langle \mathcal{T} z_i, z_i \rangle_{\mathscr{H}},$$

where the z_i are drawn from μ_δ. In practice, reasonable approximations of the trace can be obtained with a relatively small n_{tr}.

The optimization problem for finding A-optimal encoding weights is formulated as follows

$$\min_{\mathbf{w} \in \mathbf{S}} \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \langle \mathcal{C}_{\mathrm{post}}(\mathbf{w}) z_i, z_i \rangle.$$

Specializing to the cases of $\Phi_{\mathrm{GN}}(\mathbf{w})$ (2.9) and $\Phi_{\mathrm{L}}(\mathbf{w})$ (2.10) results in the following formulations,

$$\min_{\mathbf{w} \in \mathbf{S}} \left\{ \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \langle \mathcal{H}_{\mathrm{GN}}^{-1}(\mathbf{w}; m_{\mathrm{MAP}}(\mathbf{w})) z_i, z_i \rangle \right\}, \tag{2.12}$$

$$\min_{\mathbf{w} \in \mathbf{S}} \left\{ \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \langle \mathcal{H}^{-1}(\mathbf{w}; m_{\mathrm{MAP}}(\mathbf{w})) z_i, z_i \rangle \right\}. \tag{2.13}$$

Again to avoid the cost associated with the computation of the MAP point, one can evaluate the Gauss–Newton Hessian in (2.12) at a fixed reference parameter field m_0; this leads to the following (simplified) optimization problem,

$$\min_{\mathbf{w} \in \mathbf{S}} \left\{ \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \langle \mathcal{H}_{\mathrm{GN}}^{-1}(\mathbf{w}; m_0) z_i, z_i \rangle \right\}. \tag{2.14}$$

The formulation (2.14) can be seen as an extension of the formulation proposed in [48] to a fully nonlinear inverse problem formulated at the infinite-dimensional level.

2.3 Application to the Helmholtz inverse problem

In this section, we elaborate the A-optimal encoding weights formulation introduced in section 2.2 for the Helmholtz inverse problem. Recall that high resolution reconstructions in this application require a large number of experiments and that the computational cost of the inversion scales linearly with the number of experiments (see section 1.3). Source encoding can provide a trade-off between high-quality reconstruction and computational cost.

We begin by describing the inverse problem used in our study (section 2.3.1). Then the optimization problem to compute the A-optimal encoding weights, including the adjoint-based expressions for the gradient of this objective function, is detailed in section 2.3.2.

2.3.1 The inverse problem: medium parameter reconstruction

For simplicity of the presentation, we derive the formulation using a single frequency but extensions to the case of multiple frequencies are straightforward. We use homogeneous Neumann boundary conditions. The frequency-domain Helmholtz equation is given, for $i = 1, \ldots, N_w$, by

$$\begin{aligned} -\Delta u_i - \kappa^2 m u_i &= f(\mathbf{w}^i), \quad \text{in } \Omega, \\ \nabla u_i \cdot \mathbf{n} &= 0, \quad \text{on } \partial\Omega. \end{aligned} \tag{2.15}$$

Solutions u_i (2.15) are considered in $H^1(\Omega)$, i.e., the Sobolev space of functions in $L^2(\Omega)$ with square integrable weak derivatives. The original source terms are in the dual space of $H_0^1(\Omega)$, i.e., $f_j \in H^{-1}(\Omega)$. The (medium) parameter field $m \in L^\infty(\Omega)$ corresponds to the square of the slowness (or the squared inverse local wave speed) and the constant κ is the frequency of the wave (in rad/s).

2.3.1.1 MAP point

The MAP point is the solution to a deterministic inverse problem (see section 2.1.1) with the norms in the data-misfit and regularization terms weighted by the noise and prior covariance operators respectively. In particular, with a Gaussian prior $\mu_0 = \mathcal{N}(m_0, \mathcal{C}_0)$ and the norm corresponding to the inner product (2.2), we have

$$m_{\text{MAP}}(\mathbf{w}) = \arg\min_{m \in \mathbb{E}} \left\{ \frac{1}{2N_w} \sum_{i=1}^{N_w} \left\| Bu_i - \mathbf{d}(\mathbf{w}^i) \right\|^2_{\mathbf{\Gamma}^{-1}_{\text{noise}}} + \frac{1}{2} \left\| m - m_0 \right\|^2_{\mathbb{E}} \right\}, \tag{2.16}$$

where u_i solves (2.15).

To properly define the source terms f_i, appearing in the right hand-side of the forward problem, and the observation operator B, we define the mollifier $\varphi_\varepsilon(x; y)$ as follows:

$$\varphi_\varepsilon(x; y) = \frac{1}{\alpha_\varepsilon} e^{-\frac{1}{\varepsilon^2 - |x-y|^2}} \mathbb{1}_{\mathcal{B}(y,\varepsilon)}(x), \tag{2.17}$$

where $\alpha_\varepsilon = 2\pi K \varepsilon^2 e^{-1/\varepsilon^2}$, $K = \int_0^1 r e^{-1/(1-r^2)} dr$, $\mathbb{1}_{\mathcal{B}(y,\varepsilon)}$ is the indicator function for the ball of radius ε centered at y, and $0 < \varepsilon \ll 1$. This function is smooth and integrates to one. We choose each source terms f_i to be a mollifier centered at one of the N_s source locations that we denote x_i^s for $i = 1, \ldots, N_s$, i.e., $f_i(x) = \varphi_\varepsilon(x; x_i^s)$. The observation operator $B : H^1(\Omega) \to \mathbb{R}^q$ is the evaluation, at each of the receiver locations which we denote x_j^r for $j = 1, \ldots, q$, of a convolution between the solution to the forward problem u_i and a mollifier $\varphi_{\varepsilon'}(x; 0)$, i.e., $(Bu_i)_j = (u_i * \varphi_{\varepsilon'}(\cdot\,; 0))(x_j^r)$. These choices of the source terms and observation operator guarantee that the forward, adjoint, incremental forward and incremental adjoint solutions belong to $H^1(\Omega)$.

2.3.1.2 Gradient and Hessian of the inverse problem

Availability of derivatives of the function in brackets on the right hand side of (2.16) is required for the computation of m_{MAP}. The second derivative, i.e., the Hessian operator, also enters the A-optimal formulation laid down in section 2.2. We derive both gradient and Hessian following the formal Lagrangian approach (see section 1.3.1). The first-order necessary optimality condition for the MAP point is a coupled system of PDEs: Find $(m_{\text{MAP}}, \{u_i\}_i, \{p_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ such that for all variations $(\tilde{m}, \{\tilde{u}_i\}_i, \{\tilde{p}_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$

$$\begin{aligned} \langle \nabla u_i, \nabla \tilde{p}_i \rangle - \kappa^2 \langle m_{\text{MAP}}(\mathbf{w}) u_i, \tilde{p}_i \rangle - \langle f(\mathbf{w}^i), \tilde{p}_i \rangle &= 0, \ \forall i \\ \langle \nabla \tilde{u}_i, \nabla p_i \rangle - \kappa^2 \langle \tilde{u}_i, m_{\text{MAP}}(\mathbf{w}) p_i \rangle + \langle B\tilde{u}_i, Bu_i - \mathbf{d}(\mathbf{w}^i) \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} &= 0, \ \forall i \\ \langle m_{\text{MAP}}(\mathbf{w}) - m_0, \tilde{m} \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \langle u_i p_i, \tilde{m} \rangle &= 0. \end{aligned} \tag{2.18}$$

For the Hessian, we describe the solution to the equation $y = \mathcal{H}^{-1}(m_{\text{MAP}})z$. This leads to the coupled system of PDEs: Find $(y, \{v_i\}_i, \{q_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ such that for all $(\tilde{m}, \{\tilde{u}_i\}_i, \{\tilde{p}_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ the following equations are satisfied:

$$\begin{aligned} \langle \nabla v_i, \nabla \tilde{p}_i \rangle - \kappa^2 \langle m_{\text{MAP}}(\mathbf{w}) v_i, \tilde{p}_i \rangle - \kappa^2 \langle u_i y, \tilde{p}_i \rangle &= 0, \ \forall i \\ \langle \nabla \tilde{u}_i, \nabla q_i \rangle - \kappa^2 \langle \tilde{u}_i, m_{\text{MAP}}(\mathbf{w}) q_i \rangle - \kappa^2 \langle \tilde{u}_i, p_i y \rangle + \langle B\tilde{u}_i, Bv_i \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} &= 0, \ \forall i \\ \langle y, \tilde{m} \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \Big[\langle v_i p_i, \tilde{m} \rangle + \langle u_i q_i, \tilde{m} \rangle \Big] &= \langle z, \tilde{m} \rangle. \end{aligned} \tag{2.19}$$

2.3.2 The optimization problem for A-optimal encoding weights

Here we formulate the optimization problem for computing A-optimal source encoding weights for the frequency-domain seismic inverse problem

(2.15). We restrict ourselves to the case of the Laplace formulation (2.13) as the other two functionals, (2.12) and (2.14), can be treated as special cases of the Laplace formulation.

In its original format, the optimization problem for A-optimal encoding weights (2.13) is a bi-level optimization, as the MAP point is itself the solution to a minimization problem (2.3). However this is not a practical formulation to compute derivatives. We therefore reformulate (2.13) as a PDE-constrained optimization problem in which the MAP point is defined as a solution of the first-order optimality condition (2.18). The other PDE constraint is the solution to the Hessian system (2.19) along the random directions of the trace estimator, i.e., we define the objective functional for the computation of the A-optimal encoding weights by

$$\frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle y_k, z_k \rangle,$$

where z_k is a random direction for the trace estimator and $y_k = \mathcal{H}^{-1}(m_{\text{MAP}})z_k$ according to (2.19). We can then enforce these PDE constraints with Lagrange multipliers and compute derivatives of the optimization problem (2.13) using the formal Lagrangian approach. We account for the constraint on the weights through a penalty term,

$$\frac{\lambda}{2N_w} \sum_{j=1}^{N_w} \left(\|\mathbf{w}^j\|^2 - 1 \right)^2,$$

with $\lambda > 0$. Although a penalty term is not the only option, we found this relaxation of the constraint to be efficient and easy to implement.

We now present the complete formulation for (2.13). The A-optimal

encoding weights are solutions to the minimization problem

$$\min_{\mathbf{w}} \left\{ \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle y_k, z_k \rangle + \frac{\lambda}{2N_w} \sum_{j=1}^{N_w} \left(\|\mathbf{w}^j\|^2 - 1 \right)^2 \right\}, \tag{2.20}$$

where for every $k = 1, \ldots, n_{tr}$, $(y_k, \{v_{i,k}\}_i, \{q_{i,k}\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ solves the system

$$\begin{aligned}
\langle \nabla v_{i,k}, \nabla \tilde{p}_{i,k} \rangle - \kappa^2 \langle m_{\text{MAP}}(\mathbf{w}) v_{i,k}, \tilde{p}_{i,k} \rangle - \kappa^2 \langle u_i y_k, \tilde{p}_{i,k} \rangle &= 0, \ \forall i \\
\langle \nabla \tilde{u}_{i,k}, \nabla q_{i,k} \rangle - \kappa^2 \langle \tilde{u}_{i,k}, m_{\text{MAP}}(\mathbf{w}) q_{i,k} \rangle - \kappa^2 \langle \tilde{u}_{i,k}, p_i y_k \rangle & \\
+ \langle B\tilde{u}_{i,k}, B v_{i,k} \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} &= 0, \ \forall i \\
\langle y_k, \tilde{m} \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \Big[\langle v_{i,k} p_i, \tilde{m} \rangle + \langle u_i q_{i,k}, \tilde{m} \rangle \Big] &= \langle z_k, \tilde{m} \rangle,
\end{aligned} \tag{2.21}$$

for all $(\tilde{m}, \{\tilde{u}_{i,k}\}_i, \{\tilde{p}_{i,k}\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ and where $(m_{\text{MAP}}, \{u_i\}_i, \{p_i\}_i)$ $\in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ solves the first-order optimality system for the Helmholtz inverse problem

$$\begin{aligned}
\langle \nabla u_i, \nabla \tilde{p}_i \rangle - \kappa^2 \langle m_{\text{MAP}}(\mathbf{w}) u_i, \tilde{p}_i \rangle - \langle f(\mathbf{w}^i), \tilde{p}_i \rangle &= 0, \ \forall i \\
\langle \nabla \tilde{u}_i, \nabla p_i \rangle - \kappa^2 \langle \tilde{u}_i, m_{\text{MAP}}(\mathbf{w}) p_i \rangle + \langle B\tilde{u}_i, Bu_i - \mathbf{d}(\mathbf{w}^i) \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} &= 0, \ \forall i \\
\langle m_{\text{MAP}}(\mathbf{w}) - m_0, \tilde{m} \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \langle u_i p_i, \tilde{m} \rangle &= 0,
\end{aligned}$$

for all $(\tilde{m}, \{\tilde{u}_i\}_i, \{\tilde{p}_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$.

2.3.2.1 Gradient of the A-optimal weight problem

We derive the gradient of the objective function defined in (2.20), with respect to $\mathbf{w}$, using a formal Lagrangian approach. We refer the reader to appendix A.1 for this derivation. Since we enforce the PDE constraints weakly using Lagrange multipliers, we introduce adjoint variables that are indicated with

a star superscript, e.g., m^* is the adjoint variable for m. The gradient is given by $[\delta_{\mathbf{w}^1}\Phi_L(\mathbf{w}), \delta_{\mathbf{w}^2}\Phi_L(\mathbf{w}), \ldots, \delta_{\mathbf{w}^{N_w}}\Phi_L(\mathbf{w})]^T$, where for any $i = 1, \ldots, N_w$,

$$\delta_{\mathbf{w}^i}\Phi_L(\mathbf{w}) = -\frac{1}{N_w}\begin{bmatrix} \langle f_1, u_i^*\rangle & + & \langle Bp_i^*, \mathbf{d}_1\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} \\ \langle f_2, u_i^*\rangle & + & \langle Bp_i^*, \mathbf{d}_2\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} \\ & \vdots & \\ \langle f_{N_s}, u_i^*\rangle & + & \langle Bp_i^*, \mathbf{d}_{N_s}\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} \end{bmatrix}.$$

The variables u_i^* and p_i^* are computed by solving the following Hessian-like system (compare with (2.19)): Find $(m^*, \{u_i^*\}_i, \{p_i^*\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ such that for all $(\tilde{m}, \{\tilde{u}_i\}_i, \{\tilde{p}_i\}_i) \in \mathbb{E} \times H^1(\Omega)^{N_w} \times H^1(\Omega)^{N_w}$ the following equations are satisfied:

$$\begin{aligned}
\langle \nabla p_i^*, \nabla \tilde{p}_i\rangle - \kappa^2\langle m p_i^*, \tilde{p}_i\rangle - \kappa^2\langle u_i m^*, \tilde{p}_i\rangle &= -\frac{2}{n_{tr}}\sum_{k=1}^{n_{tr}} \kappa^2\langle v_{i,k} y_k, \tilde{p}_i\rangle, \\
\langle \nabla u_i^*, \nabla \tilde{u}_i\rangle - \kappa^2\langle m u_i^*, \tilde{u}_i\rangle - \kappa^2\langle p_i m^*, \tilde{u}_i\rangle & \\
+\langle Bp_i^*, B\tilde{u}_i\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} &= -\frac{2}{n_{tr}}\sum_{k=1}^{n_{tr}} \kappa^2\langle y_k q_{i,k}, \tilde{u}_i\rangle, \\
\langle m^*, \tilde{m}\rangle_{\mathbb{E}} - \frac{1}{N_w}\sum_{i=1}^{N_w} \kappa^2\left[\langle u_i u_i^*, \tilde{m}\rangle + \langle p_i^* p_i, \tilde{m}\rangle\right] &= -\frac{2}{n_{tr}N_w}\sum_{k=1}^{n_{tr}}\sum_{i=1}^{N_w} \kappa^2\langle v_{i,k} q_{i,k}, \tilde{m}\rangle.
\end{aligned} \tag{2.22}$$

The variables $\{v_{i,k}\}$ (resp. $\{q_{i,k}\}$) are the incremental state (resp. adjoint) variables which occur in the application of the inverse Hessian in the direction of the k-th trace estimator direction z_k.

2.3.2.2 Discretization

The numerical solution of (2.20) is done via the OTD approach (see section 1.3.1), where the discretization is based on continuous Galerkin finite element with Lagrange nodal basis functions. Extra care is needed for the

discretization of the covariance operator to ensure that its discrete representation faithfully represents the properties of the target infinite-dimensional object. We do not provide full details of the discretization and refer the reader to [3, 15]. However, we show how to select the discrete random directions z_k in the trace estimator. Let us call V_h the finite-dimensional approximation to the space $H^1(\Omega)$ used for the finite-element representations of all state, adjoint, corresponding incremental variables and their respective adjoints. And let V_h^m be the finite-dimensional space for the medium parameter m. Let us call $\{\psi_i\}_{i=1}^t$ (resp. $\{\phi_i\}_{i=1}^l$) a basis for V_h (resp. V_h^m). Let us introduce the vector notations $\mathbf{x}_k = (y_k^1, \dots, y_k^l)^T$ (resp. $\mathbf{z}_k = (z_k^1, \dots, z_k^l)^T$) for the finite element representations of y_k (resp. z_k) in V_h^m. The finite-dimensional approximation to the trace estimation is then

$$\frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle y_k^h, z_k^h \rangle_{L^2} = \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \sum_{i,j=1}^{l} y_k^i z_k^j \langle \phi_i, \phi_j \rangle_{L^2} = \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle \mathbf{x}_k, \mathbf{z}_k \rangle_{\mathbf{M}},$$

with $\mathbf{M}_{ij} = \langle \phi_i, \phi_j \rangle_{L^2}$ the mass matrix in V_h^m. From the definition of y_k, we see that each y_k^h solves the system $\langle \mathcal{H} y_k^h, \phi_i \rangle_{L^2} = \langle z_k^h, \phi_i \rangle_{L^2}$, for $i = 1, \dots, l$. Substituting the representation of y_k^h and z_k^h in the basis of V_h^m, we obtain the matrix system $\mathbf{H}\mathbf{x}_k = \mathbf{M}\mathbf{z}_k$, where $\mathbf{H}$ is the standard Hessian matrix obtained from finite-element discretization of system (2.19), i.e., $\mathbf{H}_{ij} = \langle \mathcal{H}\phi_j, \phi_i \rangle_{L^2}$. The finite-dimensional approximation to the trace estimation becomes

$$\frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle y_k^h, z_k^h \rangle_{L^2} = \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle \mathbf{H}^{-1}\mathbf{M}\mathbf{z}_k, \mathbf{z}_k \rangle_{\mathbf{M}} = \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle \mathbf{H}_{\mathbf{M}}^{-1}\mathbf{z}_k, \mathbf{z}_k \rangle_{\mathbf{M}},$$

where we defined $\mathbf{H}_{\mathbf{M}}^{-1} \coloneqq \mathbf{H}^{-1}\mathbf{M}$. The matrix $\mathbf{H}_{\mathbf{M}}^{-1}$ is $\mathbf{M}$-symmetric [15], i.e., self-adjoint with respect to the $\mathbf{M}$ inner-product. Then it was proved in [3] that $\frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle \mathbf{H}_{\mathbf{M}}^{-1}\mathbf{z}_k, \mathbf{z}_k \rangle_{\mathbf{M}}$ is indeed a trace estimator provided $\mathbf{z}_k \sim \mathcal{N}(0, \mathbf{M}^{-1})$. In practice, vectors $\mathbf{z}_k$ are sampled by taking draws $\mathbf{x}_k$ from multivariate standard normal distribution, $\mathbf{x}_k \sim \mathcal{N}(0, \mathbf{I})$, and using $\mathbf{z}_k = \mathbf{M}^{-1/2}\mathbf{x}_k$

2.3.2.3 Computational cost

Problem (2.20) is highly nonlinear and requires iterative methods to be solved. The gradient, derived in section 2.3.2.1, allows us to use quasi-Newton methods [67]. In table 2.1, we report the dominating terms of the computational cost of evaluating the objective function and its gradient in all three cases (2.12)-(2.14). Additionally, it is possible to reduce the cost of formulation (2.12) by computing a low-rank approximation of the Hessian operator [36]. One must keep in mind, however, that the incremental state variables $\{v_{i,k}\}$ and incremental adjoint variables $\{q_{i,k}\}$ corresponding to each random directions $\{z_k\}$ are required to compute the gradient. For this reason, a low-rank approximation of the Hessian will only lower the computational cost when $n_{tr} > n_{\text{cg}} n_{\text{newt}}$.

Table 2.1: Computational cost for objective function and gradient evaluation of the optimization problem for finding A-optimal encoding weights. We report the computational cost, in terms of the number of forward PDE solves, for $\Phi_{\text{GN}}(\mathbf{w})$, $\Phi_{\text{L}}(\mathbf{w})$, and $\Phi_0(\mathbf{w})$ defined in (2.12)–(2.14) respectively. Notations: n_{cg} = number of Conjugate-Gradient iterations to compute the search direction in Newton's method; n_{newt} = number of Newton steps to compute the MAP point.

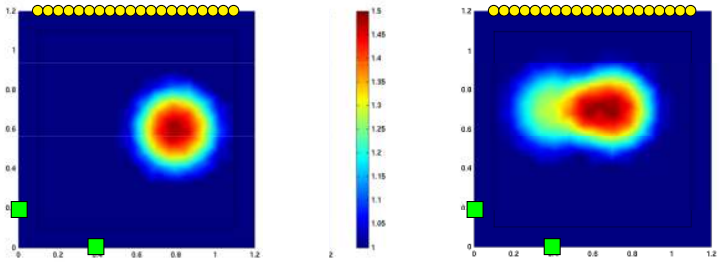
	$\Phi_0(\mathbf{w})$	$\Phi_{\text{GN}}(\mathbf{w})$ and $\Phi_{\text{L}}(\mathbf{w})$ (no low-rank)	$\Phi_{\text{GN}}(\mathbf{w})$ (with low-rank)
objective evaluation			
MAP point	$2N_w$	$2N_w n_{\text{cg}} n_{\text{newt}}$	$2N_w n_{\text{cg}} n_{\text{newt}}$
$\text{tr}(\mathcal{H}^{-1})$	$2N_w n_{\text{cg}} n_{tr}$	$2N_w n_{\text{cg}} n_{tr}$	$2N_w n_{\text{cg}}$
gradient evaluation			
v_{ik}, q_{ik}	–	–	$2N_w n_{tr}$
m^*	–	$2N_w n_{\text{cg}}$	–
u_i^*, p_i^*	N_w	–	$2N_w$
total	$2N_w n_{\text{cg}} n_{tr}$	$2N_w n_{\text{cg}}(n_{\text{newt}}+n_{tr})$	$2N_w(n_{\text{cg}} n_{\text{newt}}+n_{tr})$

Following the OTD approach, the optimization problem (2.20) is formulated in function space, before being solved with algorithms whose performance does not depend on the discretization size. This results in the overall computational cost being independent of the discretization of the parameter space, or in other words, each of the quantities n_{newt}, n_{cg} and n_{tr} in table 2.1 remain constant when the mesh gets refined. We spend the rest of this section discussing the choice of such discretization-invariant algorithms. First, we use Newton's method, with Armijo line search, to compute the MAP point; the number of Newton steps needed to converge, n_{newt}, is typically independent of the size of the parameter space [28]. Moreover, the Hessian system (2.19) needed to compute the MAP point, to evaluate the objective function (2.20), and to compute the adjoint variable m^* (2.22), is solved using the preconditioned Conjugate Gradient method [67]. The Conjugate Gradient solver is preconditioned by the prior covariance operator; the number of iterations n_{cg} needed to solve the Hessian system then depends on the spectral properties of the prior-preconditioned data-misfit part of the Hessian operator (i.e., the Hessian in function space) and is therefore independent of the discretization. The trace estimator displays a similar type of behaviour. The number of trace estimator vectors n_{tr} one should use depends on the spectral properties of the underlying infinite-dimensional operator. The choice of a discrete inner-product weighted by the mass matrix (see section 2.3.2.2) guarantees that our discrete operator will be a valid approximation of the infinite-dimensional operator and will conserve its spectral properties. The actual evaluation of the trace is performed through the repeated solution of the Hessian system (2.21), which was shown above to be discretization-independent.

2.4 Numerical results

In this section, we present numerical results for the Helmholtz inverse problem in two (spatial) dimensions. We start with a low-dimensional example ($N_w = 1$ for $N_s = 2$), which allows us to visualize the objective functions defined in section 2.2.2 over the entire weight space. This facilitates a qualitative comparison of the different approximations introduced, the Gauss–Newton (2.9) and Laplace objective functions (2.10), along with the linearized formulation (2.11). We then present an example with a higher-dimensional weight space ($N_s = 10$) in which we study the distribution of the A-optimal encoding weights and random weights sampled from the uniform spherical distribution and how the number of encoded weight vectors influence these results.

The setting for this section is a square domain with 20 receivers located at the top of the domain, and sources positioned on the bottom and left edges of the domain. The source term is a mollifier (2.17) with $\varepsilon = 10^{-6}$. This choice of source terms was numerically found to be reasonably well approximated, at the discrete level, by a point source; we utilize that approximation in this section. We use a wave frequency of $\kappa = 2\pi$ in equation (2.15). All partial differential equations are discretized by continuous Galerkin finite elements (linear elements for the parameters and quadratic elements for the state and adjoint variables). This results in a (medium) parameter space of 182 degrees of freedom. We work with synthetic data that are polluted by a 2% additive Gaussian noise.



(a) medium parameter 1 (b) medium parameter 2

Figure 2.1: Target medium parameters, along with the locations of the sources (green squares) and receivers (yellow circles).

2.4.1 One-dimensional weight space

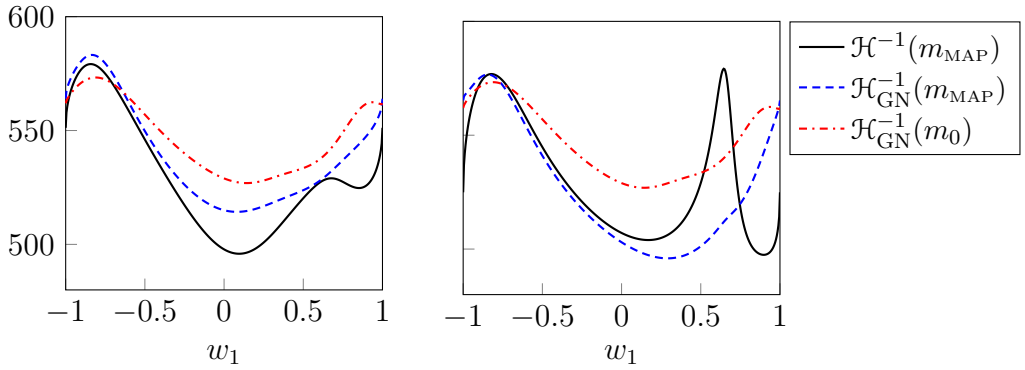
In this section, we study a one-dimensional source encoding problem corresponding to a single linear combination of two sources ($N_s = 2$ and $N_w = 1$). Although this setting represents an unrealistic situation (low number of sources, and high ratio of number of encoded sources over total number of sources), it is informative for the following reasons: (1) It provides numerical evidence of the strong and highly nonlinear dependence of the objective functions (2.9)–(2.11) on the encoding weights. (2) It demonstrates the presence of multiple local minima in the minimization problem (2.8). (3) It highlights the difference between the Gauss–Newton and Laplace formulations. The sources are located on the bottom and left edges of the domain, and we study two different medium parameters, each made of a constant background and a smooth compactly supported perturbation (see figure 2.1).

We next define the noise covariance and the prior covariance operators used in these numerical applications. Let us introduce the non-singular, positive definite, elliptic operator $\mathcal{Y} = -\gamma\Delta + \beta I$, with γ, β positive constants, I the identity operator and Δ the Laplacian operator with homogeneous Neu-

mann boundary conditions. Then we define the prior covariance operator as $\mathcal{C}_0^{-1} = \mathcal{Y} + \eta\mathcal{Y}^2$ with $\eta > 0$. One can verify that this choice of prior covariance operator is symmetric, positive definite and trace-class as long as $\gamma, \eta, \beta > 0$. The noise covariance operator for the observations is chosen to be a multiple of the identity matrix, i.e., $\mathbf{\Gamma}_{\text{noise}} = \sigma^2\mathbf{I}$—in our examples we choose $\sigma = 1$. The parameters γ, β, and η are chosen as $\gamma = 10^{-3}$, $\beta = 10^{-4}$ and $\eta = 10^{-2}$, and we have verified that this choice approximately satisfies the discrepancy principle. In the (discrete) numerical applications, we use $\delta = 0$ in the measure μ_δ the trace estimator vectors z_i are sampled from (see section 2.2.2).

To enforce the constraint $\mathbf{w} \in \mathbf{S}$, i.e., $\sqrt{w_1^2 + w_2^2} = 1$ in this case, we parameterize the weight vector as $(w_1, \pm\sqrt{1 - w_1^2})$. The parameter w_1, alone, controls the combination of both sources. Moreover, the weight vectors $(w_1, -\sqrt{1 - w_1^2})$ and $(-w_1, \sqrt{1 - w_1^2})$ lead to the same reconstruction, such that it suffices to consider the encoding weights $(w_1, \sqrt{1 - w_1^2})$ for $w_1 \in [-1, 1]$.

In figure 2.2, we plot the three objective functions (2.9)–(2.11) from section 2.2.2. For each $w_1 \in [-1, 1]$, the Gauss–Newton (2.9) and Laplace (2.10) formulations are evaluated at the MAP point, $m_{\text{MAP}}(w_1)$, corresponding to the encoding weight $(w_1, \sqrt{1 - w_1^2})$; in other words, the Hessian for these two criteria is evaluated at a medium parameter $m_{\text{MAP}}(w_1)$ that varies with the weight w_1. For formulation (2.11), we choose m_0 to be a constant value equal to the background medium, i.e., $m_0 \equiv 1$. We observe that the result for the Gauss–Newton formulation (2.9) differs from the Laplace approximation (2.10). In addition, we clearly observe that each formulation contains local minima.



(a) medium parameter 1 (b) medium parameter 2

Figure 2.2: Plots of $\operatorname{tr}(\mathcal{H}^{-1})$ with $\mathcal{H}^{-1}(m_{\text{MAP}}(w_1))$, $\mathcal{H}^{-1}_{\text{GN}}(m_{\text{MAP}}(w_1))$ and $\mathcal{H}^{-1}_{\text{GN}}(m_0)$ for both target media. $m_0 \equiv 1$, same as the background value for the medium parameter.

Robustness of the Gauss–Newton formulation (2.9) Since the computation of the MAP point $m_{\text{MAP}}(w_1)$ is a computationally intensive task for large-scale problems, it might be useful to solve the optimization (2.8) without having to recompute the exact MAP point for each iterate of the weights. The Laplace formulation (2.10) is based on the full Hessian which is guaranteed to be positive definite only in a neighbourhood of the MAP point. The Gauss–Newton approximation, however, is always positive definite and we observe numerically that it preserves relevant information about the objective function, even far away from the MAP point. In figure 2.3, we plot the objective function (2.9), for all values of $w_1 \in [-1, 1]$, for different (fixed) medium parameters $\bar{m}_s$ ranging from the background medium, $m_0 \equiv 1$, to the MAP point $m^\sharp$ computed using both sources independently (for medium parameter 2). The sources are located at the points $(0, 0.1)$ and $(0, 1.1)$. That is, we define

$$\bar{m}_s = (1 - s)m_0 + s\, m^\sharp.$$

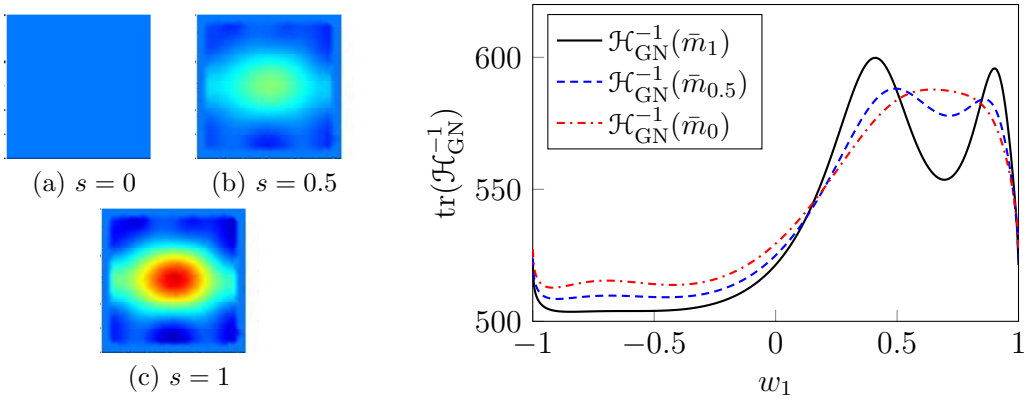


Figure 2.3: Plots of objective function Φ_0 (2.11) for weights $w_1 \in [-1, 1]$ (right), at medium $\bar{m}_s$, with $s = 0$, 0.5, 1 (left). Here $m_0 \equiv 1$ (the background medium).

It appears that the medium parameter needs to include the main features of the target medium sufficiently accurately ($s > 0.5$) to match the main features of the exact trace of the posterior covariance; this can be seen from the behavior of $\mathrm{tr}\big(\mathcal{H}_{\mathrm{GN}}^{-1}(w_1, \bar{m}_s)\big)$ in the interval $w_1 \in [0.2, 1.0]$.

The effect of trace estimation When computing A-optimal encoding weights, one only needs the local minima of the trace to be well characterized. We show in figure 2.4 that trace estimation does indeed affect the shape of the objective function in the formulations of the A-optimal encoding weights (2.13). However, in our example, the objective function using a trace estimation preserves the local minima of the objective function using an exact trace when a sufficient number of trace estimator vectors are used.

2.4.2 A-optimal encoding weights in higher dimensional weight spaces

We now consider a problem with 10 sources (i.e., $N_s = 10$). Here,

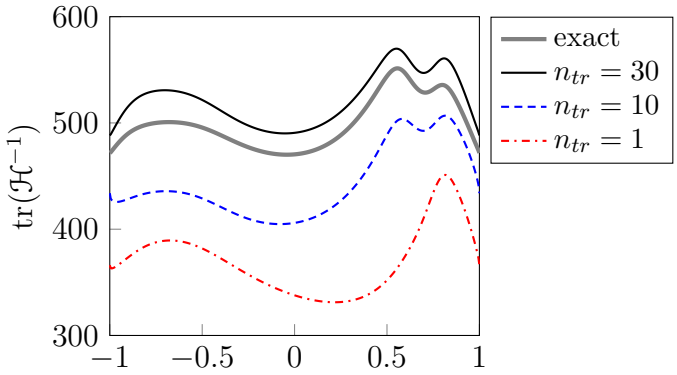


Figure 2.4: Plots of the objective function in (2.13) when the trace of the posterior covariance is computed exactly or with a trace estimator ($n_{tr} = 1, 10, 30$). For each n_{tr}, we used a fixed realization of the trace estimator vectors.

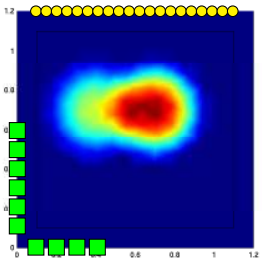
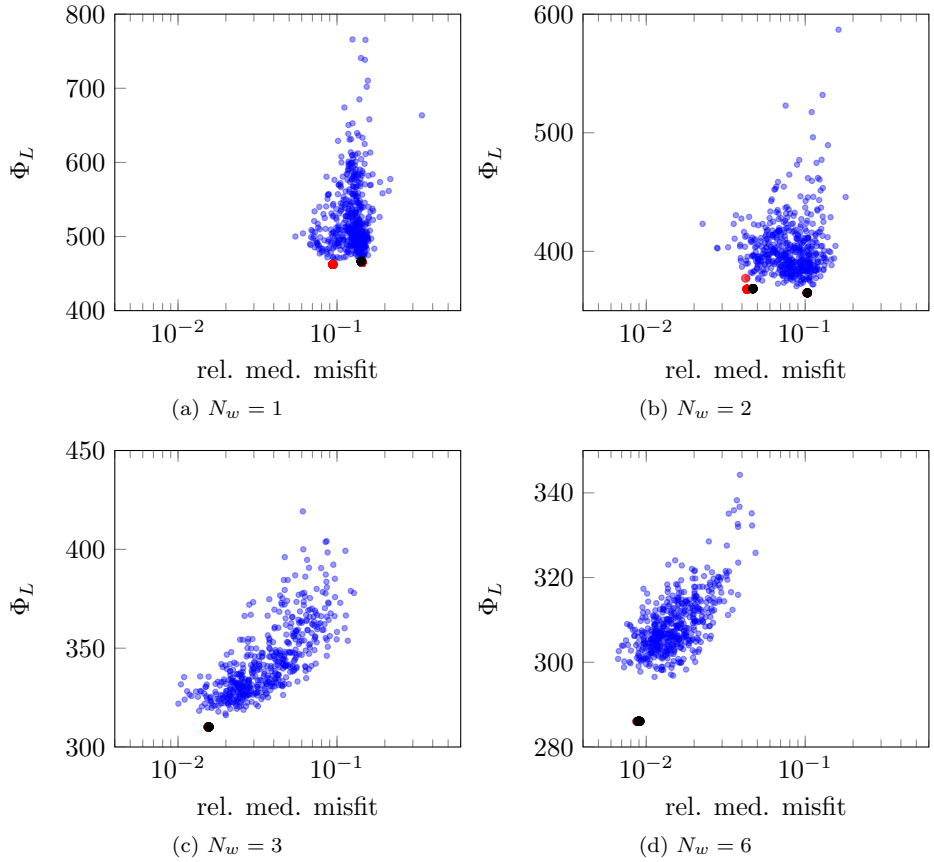


Figure 2.5: Target medium parameter and locations of the 10 sources (green squares), and receivers (yellow circles).

we focus on qualitative properties of the A-optimal source encoding weights by performing statistical tests, in which we study how successful A-optimal encoding weights are in reducing posterior variance and relative medium misfit compared to encoding weights sampled from the uniform spherical distribution. We also compared with random weights sampled, then re-scaled, from the Rademacher distribution (see section 1.5.1). Since the results we obtained were not statistically different from the results presented in this section using random weights sampled from the uniform spherical distribution, we decided

to omit these results. Throughout this section, the relative medium misfit is taken to be the relative L^2-error between the reconstruction of interest and the reconstruction obtained using all 10 sources independently. The penalty parameter was empirically selected to be $\lambda = 10^3$.



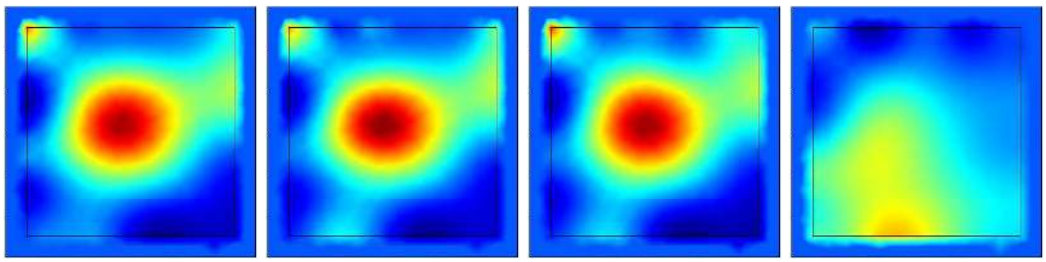
(a) $N_w = 1$ (b) $N_w = 2$

(c) $N_w = 3$ (d) $N_w = 6$

Figure 2.6: Plot of $\Phi_L(\mathbf{w})$ (2.10) against relative medium misfit ($N_s = 10$ and $N_w = 1, 2, 3, 6$) for reconstructions using random encoding sources sampled from the uniform spherical distribution (blue) or A-optimal encoding weights computed with formulation (2.12) (black) and (2.13) (red). Target model 2 with source configuration as shown in figure 2.5. Sample size = 500, and n_{tr} = 30.

We show the results in figure 2.6. Each plot shows, for different number

of encoded sources (N_w = 1, 2, 3 and 6), the objective function $\Phi_L(\mathbf{w})$ defined in (2.10) against the relative medium misfit of the reconstruction, which is an indication for the quality of the reconstruction. Each reconstruction is indicated by a translucent dot; a darker shade indicates a higher concentration of reconstructions in that part of the plot. This shows the variation in the quality of the reconstruction. The blue dots correspond to reconstructions that use random encoding weights sampled from the uniform spherical distribution. The red dots indicate A-optimal encoding weights based on the Laplace formulation (2.13). The reconstructions marked with black dots use A-optimal encoding weights based on the Gauss–Newton formulation (2.12). In order to detect potential local minima, the A-optimal encoding weights are re-computed several times, starting from different initial conditions.



(a) (b) (c) (d)

Figure 2.7: Four examples of reconstructions using different number of sources, with target parameter 2: (a) 10 independent sources; (b) 3 A-optimally encoded sources; (c) 3 randomly encoded sources; (d) 3 other randomly encoded sources.

Notice that with one encoded source, A-optimal encoding weights do not provide a clear advantage over random weights. The overall distribution of random weights does not indicate a strong connection between the trace of the posterior covariance (2.10) and the relative medium misfit. On the other hand, the A-optimal encoding weights outperform the random weights (on

average), when sufficiently many encoding weights are used (see in particular $N_w = 2$ and 3 in figure 2.6). In that case, the random weights appear to indicate a linear correlation between our objective function and the relative medium misfit, which translates into the best reconstruction being also the one with smallest trace of the posterior covariance. Overall, these results suggest the existence of a threshold, in the number of encoding sources, above which optimal weights provide improvement in both variance and medium misfit over random encoding weights. Moreover, based on these results, there does not appear to be a clear advantage in using the Laplace approximation (2.13) over the Gauss–Newton approximation (2.12), provided sufficiently many encoded sources are used. In the last row of figure 2.6, optimal weights computed with both formulations provide similar results, although the actual values of the weights do not necessarily agree.

In addition, we provide a comparison of the reconstructions computed using all sources independently (figure 2.7a), using three A-optimally encoded sources (figure 2.7b), and two examples of reconstructions computed using three randomly encoded sources: one resulting in a good reconstruction (figure 2.7c), and one resulting in a poor reconstruction (figure 2.7d). There is virtually no difference between the reconstructions computed using all 10 sources and using three A-optimally encoded sources. On the other hand, using random encoding weights drawn from the same distribution may lead to good or poor reconstructions, as is shown in figures 2.7c, d. This is consistent with the results in figure 2.6 (bottom left), where the blue dots show large variations in terms of relative medium misfit.

Variability of the A-optimal encoding weights The A-optimal encoding weight formulation introduced in section 2.2 relies on a fixed realization of the trace estimator vectors. Note that the A-optimal encoding weights are solutions to a highly nonlinear optimization problem that in general exhibits local minima. However, we show numerically that, provided sufficiently many encoding weights are chosen and a large enough number of trace estimator vectors are used, the computation of the A-optimal encoding weights is stable with respect to trace estimation. In figure 2.8, we show 100 results obtained

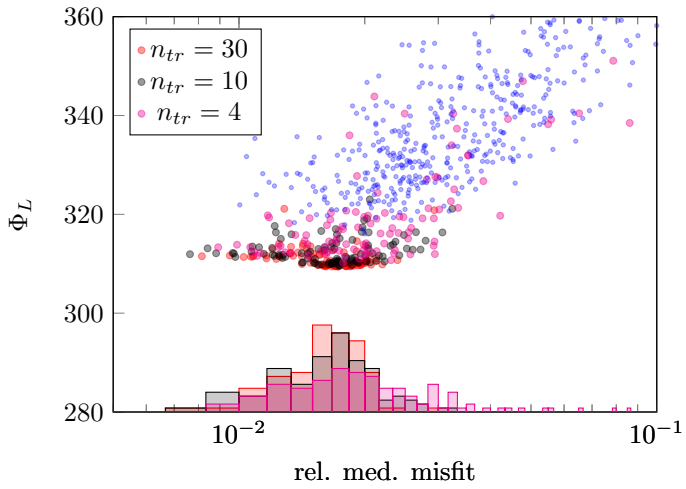


Figure 2.8: Variability of the A-optimal weights for different numbers of trace estimator vectors, $n_{tr} = 30$ (red), 10 (black) and 4 (magenta). A-optimal encoding weights are computed with formulation (2.13) ($N_s = 10$ and $N_w = 3$), using different realizations of the trace estimator vectors and different initial guess of the weights. Sample size = 100.

with Laplace A-optimal encoding weights (2.13), in the case of 3 encoded sources, with different numbers of trace estimator vectors ($n_{tr} = 4$, 10, 30). Each computation uses different realizations of the trace estimator vectors, and different initial guess of the weights.

We observe that with $n_{tr} = 10$ and 30 the computations of the A-optimal encoding weights provide similar results. On the other hand, the use of 4 trace estimator vectors leads to a much wider range in the quality of the results, both in terms of relative medium misfit and trace of the posterior covariance.

2.4.3 Remarks on the Gauss–Newton formulation

Here, we discuss the justification for and advantages of using the Gauss–Newton formulation for finding A-optimal encoding weights. In many important situations, the Gauss–Newton formulation appears accurate enough to compute the A-optimal encoding weights. The Gauss–Newton approximation to the Hessian is most accurate when the data misfit residual is small at the solution of the inverse problem. This is the case, for instance, when the noise level in the observations is low. In our numerical experiments we observed that, provided sufficiently many encoded sources are used, the Gauss–Newton formulation represents a sufficiently accurate approximation to the Laplace formulation for the purpose of computing A-optimal encoding weights.

The Gauss–Newton formulation holds strong promises to reduce the computational cost of the A-optimal encoding weights. The data-misfit part of the Gauss–Newton Hessian is guaranteed to be positive semi-definite at any evaluation point, and hence the Gauss–Newton Hessian is positive definite. This allows two main improvements to the computations of the A-optimal weights. First, and as detailed in section 2.3.2.3, one can incorporate a low-rank approximation of the Gauss–Newton Hessian to reduce the computational cost. The magnitude of that reduction is problem-dependent, but will be most noticeable when large numbers of trace estimator vectors are required.

Another advantage of the positive definiteness of the Gauss–Newton Hessian is that the objective function (2.9) of the Gauss–Newton formulation does not have to be evaluated in a small neighbourhood of the MAP point for the objective function to make sense. This could allow one, for instance, to solve the MAP point inexactly when the A-optimal objective function is far from its minimum, which would reduce the overall computational cost. In section 2.4.1, we studied how the objective function varies with the evaluation point $\bar{m}_s$ (figure 2.3), and observed that the objective function tends to maintain similar local minima away from the MAP point.

Finally, we want to point out that in certain situations, the full Hessian may not be available, may be too complicated to derive, or too expensive to compute, rendering the Laplace formulation inadequate. This can be the case for inverse problems with highly nonlinear forward problems.

2.5 Conclusion

We have developed a method for the computation of A-optimal encoding weights aiming at large-scale non-linear inverse problems. As we show numerically, reconstructions obtained using A-optimal encoding weights not only minimize the average of the posterior variance, but consistently outperform random encoding weights in terms of the quality of the reconstructions. While in this work, we relied on quasi-Newton methods for solving the optimization problem for A-optimal encoding weights, we will explore the derivation and implementation of a Newton solver for this optimization problem in future work. We point out that, thanks to the optimize-then-discretize approach we adopted, the derivation of the analytical expression for the action of the Hessian in a direction is possible with little more effort than what was

required to get the gradient.

We introduced two formulations for the computation of the A-optimal encoding weights, namely the Gauss–Newton formulation (2.12) and the Laplace formulation (2.13). Although the Gauss–Newton formulation represents an approximation to the Laplace formulation, it holds several advantageous features from computational point of view.

We note that computing A-optimal encoding weights can entail a significant computational effort. However, the method can be attractive for real-time monitoring applications where one needs to solve an inverse problem repeatedly over time. In this case, one first computes the A-optimal encoding weights offline, and then can use those weights to solve the inverse problem repeatedly at a fraction of the original cost. An example for such an application is the monitoring of an oil reservoir, where seismic or electro-magnetic inverse problems are solved repeatedly to characterize the evolution of the reservoir properties over time.

Chapter 3

A comparative study of regularizations for multi-parameter inverse problems

We now turn our attention to multi-parameter and multi-physics inverse problems. In this chapter, we investigate what joint regularization terms are both, amenable for large-scale applications, and best able to reconstruct sharp edges in the truth parameter fields. Indeed, truth parameter fields in geophysical and medical applications often present sharp variations in contrast, which are of critical importance for diagnostic and analysis. Based on these criteria and a current literature review, we shortlisted the following four joint regularization terms: the cross-gradient paired with independent TV regularizations, the normalized cross-gradient paired with independent TV regularizations, the vectorial total variation joint regularization, and a novel nuclear norm joint regularization that we introduce in section 3.3. Note that in [44], the authors compared a number of joint regularizations, including the cross-gradient and VTV, but our study updates the candidate list of the joint regularizations, compare them over a more diverse range of examples, and put the emphasis on reconstructing sharp interfaces, a challenge for numerical applications.

The main contributions of this chapter are as follows: (1) We carry out a detailed comparison of the four joint regularization terms we identified, within the computational framework of large-scale inverse problems (sec-

tion 1.3). We illustrate the strengths and weaknesses of all four regularizations through a comprehensive numerical study that covers a range of practical situations. (2) We introduce a novel joint regularization term, for joint inverse problems constrained by PDEs, that promote gradient alignment through the use of the nuclear norm. Although a similar idea was used previously in medical imaging [58], this is, to the best of our knowledge, the first time such a regularization is used in the context of PDE-constrained inverse problems. (3) We demonstrate numerically how a Poisson inverse problem can supply low-frequency information to an inverse problem governed by the acoustic wave equation through the formulation of a multi-physics inverse problem.

The first three sections of this chapter are dedicated to introducing the four joint regularization terms whose performance we will compare in section 3.4. The cross-gradient and normalized cross-gradient are discussed in sections 3.1.1 and 3.1.2. The vectorial total variation is presented in section 3.2, and the nuclear norm joint regularization is defined and discussed in section 3.3. The numerical section 3.4 presents results from three different examples of joint inverse problems. In section 3.4.1, we use two joint Poisson inverse problems to illustrate some key features of each structural similarity term. We then study, in section 3.4.2, how joint inversion can improve the reconstructions of the bulk modulus and the density in the acoustic wave equation. In section 3.4.3, we look at a multi-physics joint inverse problem, combining a Poisson inverse problem and an acoustic wave inverse problem. We provide some concluding remarks in section 3.5.

Although this is not a limitation of the methods we present, for the sake of brevity, we restrict ourselves to two inversion parameters in this chapter; We consider larger number of inversion parameters in chapter 4. In sections 3.1

to 3.3, we denote the inversion parameters by m_1 and m_2. In this chapter, we make extensive use of the notation for the modified TV, $\mathcal{R}_{\text{TV},\varepsilon}$ (1.6), introduced in section 1.2.2.1.

3.1 Cross-gradient terms

In this section, we introduce the cross-gradient term and its normalized version. The main idea behind both structural similarity terms is to force the level sets of the inversion parameters m_1 and m_2 to align. As illustrated in figure 3.1, alignment of the level sets is equivalent to the alignment of the gradients of the fields m_1 and m_2 at each point. By definition of the cross-

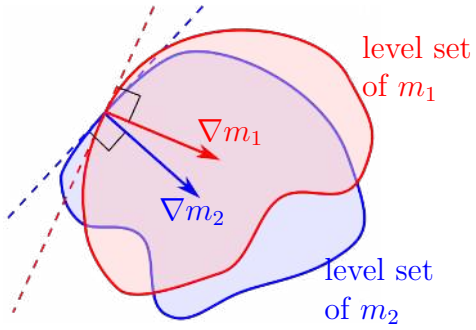


Figure 3.1: Sketch of a level set for the parameters m_1 (red) and m_2 (blue), with their respective gradients at a point.

product of two vectors, the gradients of two parameters m_1 and m_2 are aligned, at a given point, when $|\nabla m_1 \times \nabla m_2|^2$ vanishes.

3.1.1 The cross-gradient term

The cross-gradient term $\hat{\mathcal{R}}_{\text{cg}}$, defined as

$$\hat{\mathcal{R}}_{\text{cg}}(m_1, m_2) \coloneqq \frac{1}{2}\int_\Omega |\nabla m_1 \times \nabla m_2|^2 \, dx, \tag{3.1}$$

was introduced in [37] and has become a popular choice in geophysical applications, particularly in seismic imaging. Although the formulation (3.1) is intuitive, it is not convenient for numerical discretization and to compute derivatives. Using vector calculus, we re-write (3.1) as

$$\hat{\mathcal{R}}_{\text{cg}}(m_1, m_2) = \frac{1}{2}\int_\Omega |\nabla m_1|^2 |\nabla m_2|^2 - (\nabla m_1 \cdot \nabla m_2)^2 \, dx. \tag{3.2}$$

Combining the cross-gradient term (3.2) with independent TV regularizations for m_1 and m_2, we obtain the joint regularization

$$\mathcal{R}(m_1, m_2) = \gamma_1 \mathcal{R}_{\text{TV},\varepsilon}(m_1) + \gamma_2 \mathcal{R}_{\text{TV},\varepsilon}(m_2) + \gamma \hat{\mathcal{R}}_{\text{cg}}(m_1, m_2), \tag{3.3}$$

where $\gamma, \gamma_1, \gamma_2 > 0$. In [44] the authors propose a different formulation, in which each independent TV regularization is weighted by a non-linear function of the gradient of the other parameter. The goal of this weighting is to apply TV regularization only for points in the parameter space where the cross-gradient term alone is not sufficient to prevent oscillatory solutions. Such oscillations may occur where the gradient of one parameter is very small, resulting in an (almost) vanishing cross-gradient term. Because this formulation further increases the nonlinearity of the problem (nonlinear inverse problem, nonlinear regularization, nonlinear structural similarity term), we instead use (3.3).

Derivatives of the cross-gradient structural similarity term can provide insight about its action on the reconstructions of the joint inverse problem. Using formulation (3.2), we compute the gradient, and the action of the Hessian in a given direction, for the cross-gradient term [26]. The weak form of the gradient, evaluated at a point $m \coloneqq (m_1, m_2)$, and acting in a direc-

tion $\tilde{m} \coloneqq (\tilde{m}_1, \tilde{m}_2)$ is given by

$$\delta_{m_1}\hat{\mathcal{R}}_{\text{cg}}(m;\tilde{m}_1) = \int_\Omega |\nabla m_2|^2(\nabla\tilde{m}_1\cdot\nabla m_1) - (\nabla m_1\cdot\nabla m_2)(\nabla\tilde{m}_1\cdot\nabla m_2)\,dx,$$
$$\delta_{m_2}\hat{\mathcal{R}}_{\text{cg}}(m;\tilde{m}_2) = \int_\Omega |\nabla m_1|^2(\nabla\tilde{m}_2\cdot\nabla m_2) - (\nabla m_1\cdot\nabla m_2)(\nabla\tilde{m}_2\cdot\nabla m_1)\,dx.$$

Then the weak form of the action of the Hessian of the cross-gradient term, at a point m, along a direction $\hat{m} = (\hat{m}_1, \hat{m}_2)$, is given block-wise by

$$\delta^2_{m_1}\hat{\mathcal{R}}_{\text{cg}}(m;\hat{m}_1,\tilde{m}_1) = \int_\Omega |\nabla m_2|^2(\nabla\tilde{m}_1\cdot\nabla\hat{m}_1) - (\nabla\tilde{m}_1\cdot\nabla m_2)(\nabla m_2\cdot\nabla\hat{m}_1)\,dx,$$
$$\begin{aligned}\delta^2_{m_1,m_2}\hat{\mathcal{R}}_{\text{cg}}(m;\hat{m}_2,\tilde{m}_1) = \int_\Omega\ & 2(\nabla\tilde{m}_1\cdot\nabla m_1)(\nabla m_2\cdot\nabla\hat{m}_2)\\ & - (\nabla m_1\cdot\nabla m_2)(\nabla\tilde{m}_1\cdot\nabla\hat{m}_2) - (\nabla\tilde{m}_1\cdot\nabla m_2)(\nabla m_1\cdot\nabla\hat{m}_2)\,dx,\end{aligned}$$
$$\delta^2_{m_2}\hat{\mathcal{R}}_{\text{cg}}(m;\hat{m}_2,\tilde{m}_2) = \int_\Omega |\nabla m_1|^2(\nabla\tilde{m}_2\cdot\nabla\hat{m}_2) - (\nabla\tilde{m}_2\cdot\nabla m_1)(\nabla m_1\cdot\nabla\hat{m}_2)\,dx.$$

The Hessian provides insight about the shape of the objective function at the minimizer, where the gradient vanishes. It also affects the Newton search direction through its inverse. The weak form of the Hessian is, however, not convenient to analyze its action. In strong form, the Hessian $\mathcal{H}$ acts, in a direction $\hat{m}$, like a vector anisotropic diffusion operator, i.e.,

$$\mathcal{H}\hat{m} = -\nabla\cdot(A_{\text{cg}}(m)\nabla\hat{m}),$$

where A_{cg} is a diffusion tensor. The diffusion tensor controls the diffusive action of the Hessian operator. In particular, the kernel of the diffusion tensor contains search directions that will not be smoothed out, hence potentially preserving sharp edges.

This diffusion tensor, in the case of the cross-gradient (3.2), is given by

$$A_{\text{cg}}(m) = \begin{bmatrix} D(m_2) & B(m) \\ B(m)^T & D(m_1) \end{bmatrix}, \tag{3.4}$$

where we introduced the notations

$$D(m_i) \coloneqq |\nabla m_i|^2 I - \nabla m_i \nabla m_i^T,$$
$$B(m) \coloneqq 2\nabla m_1 \nabla m_2^T - (\nabla m_1 \cdot \nabla m_2) I - \nabla m_2 \nabla m_1^T.$$

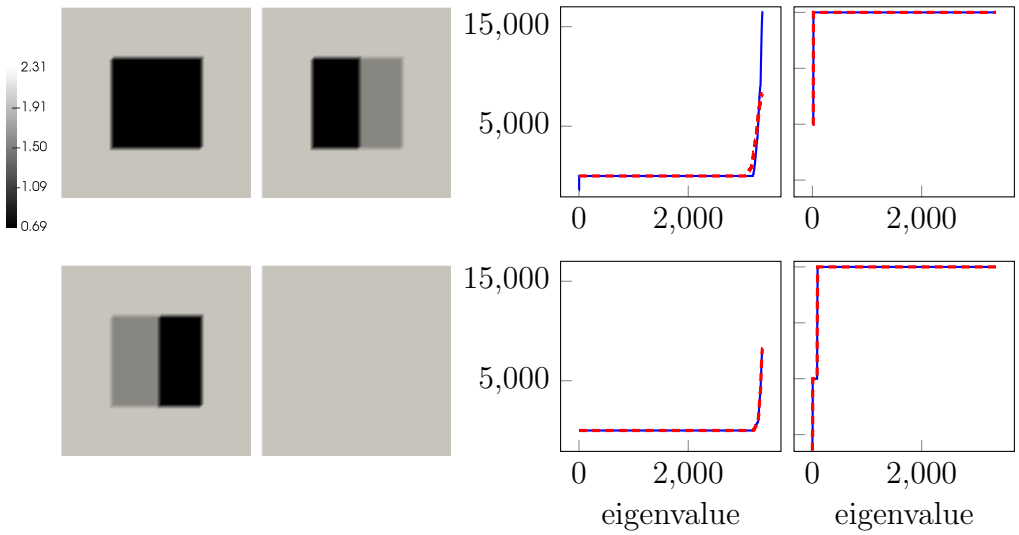
This should be compared with the diffusion tensor of the Hessian of the TV regularization for a parameter m_i, which is given by

$$A_{\text{TV}}(m_i) = \frac{1}{|\nabla m_i|}\left(I - \frac{\nabla m_i \nabla m_i^T}{|\nabla m_i|^2}\right), \tag{3.5}$$

or equivalently, $D(m_i) = |\nabla m_i|^3 A_{\text{TV}}(m_i)$. The block-diagonal part of A_{cg} indicates a TV-like behaviour but where parameter m_1 (resp. m_2) preserves sharp edges in directions where parameter m_2 (resp. m_1) presents sharp edges; this illustrates the coupling between both parameters. As we show numerically in figure 3.2, because of the presence of the off-diagonal blocks B in A_{cg}, the Hessian of the cross-gradient term can be indefinite. The TV regularization being a convex operator, its Hessian is guaranteed to be positive semidefinite. Therefore the Hessian obtained by keeping the block diagonal parts of the diffusion tensor (3.4), i.e., $\mathcal{H}_d \hat{m} \coloneqq -\nabla \cdot (A_{\text{cg},d}(m) \nabla \hat{m})$, with

$$A_{\text{cg},d}(m) \coloneqq \begin{bmatrix} D(m_2) & 0 \\ 0 & D(m_1) \end{bmatrix}, \tag{3.6}$$

is also guaranteed to be positive semidefinite. For this reason, when using the cross-gradient paired with two independent TV regularizations, we precondition the Newton system with a block-diagonal matrix containing the Hessian of the independent TV regularizations, plus a small fraction of the mass matrix in each block, plus the block-diagonal part of the Hessian of the cross-gradient term (3.6).



(i) m_1 (ii) m_2 (iii) cross-grad. (iv) norm. cr-gd

Figure 3.2: Eigenvalues of the Hessian operator (blue) and block-diagonal part of the Hessian operator (red) for the (iii) cross-gradient term (3.1) and the (iv) normalized cross-gradient term (3.7) with $\varepsilon = 10^{-4}$, for two combinations of truth parameter fields (i) m_1 and (ii) m_2. The domain is a unit square discretized by a 40×40 mesh, and the parameter fields m_1 and m_2 are discretized using continuous piecewise linear finite elements.

3.1.2 Normalized cross-gradient

A disadvantage of the cross-gradient term (3.2) is that it is minimum where one of the inversion parameters, or both, are constant, hence potentially ignoring sharp discontinuities. A workaround, first introduced for image segmentation [47], is to normalize the gradient of both inversion parameters in the formulation of the cross-gradient. The normalized cross-gradient is given by

$$\hat{\mathcal{R}}(m_1, m_2) = \frac{1}{2}\int_\Omega \left| \frac{\nabla m_1}{|\nabla m_1|} \times \frac{\nabla m_2}{|\nabla m_2|} \right|^2 dx = \frac{1}{2}\int_\Omega 1 - \left(\frac{\nabla m_1 \cdot \nabla m_2}{|\nabla m_1||\nabla m_2|} \right)^2 dx.$$

Since this formulation is non-differentiable where $|\nabla m_1| = 0$ or $|\nabla m_2| = 0$, we introduce the modified normalized cross-gradient,

$$\hat{\mathcal{R}}_{\text{ncg}}(m_1, m_2) \coloneqq \frac{1}{2}\int_\Omega 1 - \left(\frac{\nabla m_1 \cdot \nabla m_2}{\sqrt{|\nabla m_1|^2 + \varepsilon}\sqrt{|\nabla m_2|^2 + \varepsilon}}\right)^2 dx, \tag{3.7}$$

with $\varepsilon > 0$. Pairing the normalized cross-gradient term (3.7) with two TV regularizations, we obtain the joint regularization

$$\mathcal{R}(m_1, m_2) = \gamma_1 \mathcal{R}_{\text{TV},\varepsilon}(m_1) + \gamma_2 \mathcal{R}_{\text{TV},\varepsilon}(m_2) + \gamma \hat{\mathcal{R}}_{\text{ncg}}(m_1, m_2), \tag{3.8}$$

where $\gamma, \gamma_1, \gamma_2 > 0$. Unlike for the cross-gradient term, the derivatives of the normalized cross-gradient term give little insight about its regularization action. Instead, we illustrate numerically that the normalized cross-gradient often behaves as a concave operator. In figure 3.2, we plot the eigenvalues of its full Hessian and of the block-diagonal part of its Hessian for different parameter fields m_1 and m_2, and observe that most eigenvalues are negative. The main practical consequence of this observation is that the Hessian of the joint regularization (3.8) may be indefinite. For this reason, the preconditioner for the Newton system is formed by the Hessians of the independent TV regularizations alone.

3.2 Vectorial total variation

The vectorial total variation functional [14], or color TV [12], is the multi-parameter equivalent of the total variation functional. It was first introduced for multi-channel imaging applications [12, 14], and later applied to joint inverse problems [44]. Unlike the cross-gradient and normalized cross-gradient, VTV acts as a regularization in itself and does not require additional

regularization terms. It is given by

$$\mathcal{R}(m_1, m_2) = \gamma \int_\Omega \sqrt{|\nabla m_1|^2 + |\nabla m_2|^2}\, dx, \tag{3.9}$$

with $\gamma > 0$. Formulation (3.9) is non-differentiable where $|\nabla m_1| = |\nabla m_2| = 0$. For this reason, we introduce a modified VTV regularization given by

$$\mathcal{R}_{\text{VTV}}(m_1, m_2) \coloneqq \gamma \int_\Omega \sqrt{|\nabla m_1|^2 + |\nabla m_2|^2 + \varepsilon}\, dx, \tag{3.10}$$

with $\varepsilon, \gamma > 0$. Whereas the cross-gradient terms (see section 3.1) work by aligning the level sets of both inversion parameters, VTV appears to favors superimposition of discontinuities in both parameter fields. A simple, intuitive way to explain this, given the understanding of the TV regularization [18], is sketched in figure 3.3. Given two parameter fields with a single jump of same amplitude, the VTV functional is minimum when both jumps occur at the same location. This informal argument can be made rigorous by using piecewise linear functions for m_1 and m_2 instead of the Heaviside function in figure 3.3.

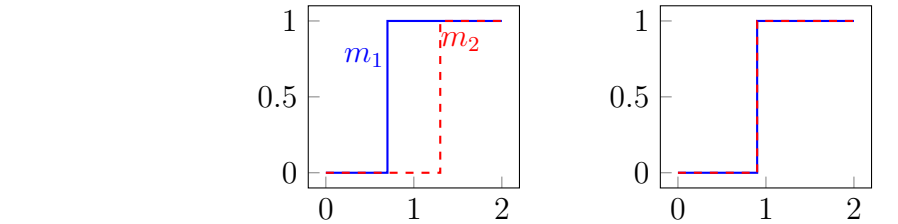


Figure 3.3: Values of the VTV regularization (3.9), for two parameter fields m_1 and m_2 defined over $\Omega = [0, 2]$, with both parameter fields having a single jump of the same amplitude, and $\mathcal{R}_{\text{TV}}(m_1) = \mathcal{R}_{\text{TV}}(m_2)$. This informal argument can be made rigorous by using piecewise linear functions for m_1 and m_2.

The derivatives of the VTV regularization resemble those of the TV regularization. For clarity, we set $\gamma \equiv 1$ in the rest of section 3.2. The weak

form of the gradient, evaluated at a point $m = (m_1, m_2)$, and acting in a direction $\tilde{m} = (\tilde{m}_1, \tilde{m}_2)$, is given by

$$\begin{aligned} \delta_{m_1}\mathcal{R}_{\text{VTV}}(m;\tilde{m}_1) &= \int_\Omega \frac{\nabla m_1 \cdot \nabla \tilde{m}_1}{\sqrt{|\nabla m_1|^2 + |\nabla m_2|^2 + \varepsilon}}\, dx \\ \delta_{m_2}\mathcal{R}_{\text{VTV}}(m;\tilde{m}_2) &= \int_\Omega \frac{\nabla m_2 \cdot \nabla \tilde{m}_2}{\sqrt{|\nabla m_1|^2 + |\nabla m_2|^2 + \varepsilon}}\, dx, \end{aligned} \tag{3.11}$$

The diffusion tensor for the Hessian of the VTV regularization, in strong form (see section 3.1.1), is given by

$$A_{\text{VTV}}(m) := \frac{1}{|\nabla m|_\varepsilon} \begin{bmatrix} I - \frac{\nabla m_1 \nabla m_1^T}{|\nabla m|_\varepsilon^2} & -\frac{\nabla m_1 \nabla m_2^T}{|\nabla m|_\varepsilon^2} \\ -\frac{\nabla m_2 \nabla m_1^T}{|\nabla m|_\varepsilon^2} & I - \frac{\nabla m_2 \nabla m_2^T}{|\nabla m|_\varepsilon^2} \end{bmatrix}, \tag{3.12}$$

where $|\nabla m|_\varepsilon^2 = |\nabla m_1|^2 + |\nabla m_2|^2 + \varepsilon$. Comparing with the diffusion tensor for the Hessian of the TV regularization (3.5), we find similar terms along the diagonal, to the exception of the normalization factor at the denominator; it is $|\nabla m_i|$ in the case of TV, and $|\nabla m|_\varepsilon$ in the case of VTV, i.e., it involves the gradient of both parameters, hence introducing coupling between the parameter fields. Since the focus of the current chapter is on a qualitative comparison of several joint regularization terms, we postpone further discussion about the VTV functional until chapter 4. A more detailed analysis of the VTV functional is carried out in section 4.1. In section 4.2, we introduce a primal-dual Newton method for the efficient, scalable solution of PDE-constrained joint inverse problems regularized with VTV.

3.3 Nuclear norm joint regularization

In [58], the authors extended the total generalized variation to vector-valued parameters, and by doing so, employed the pointwise nuclear norm of a matrix field to force alignment of image edges. Building on their work,

we propose a joint regularization utilizing the nuclear norm. As the methods discussed in section 3.1, this term aims at promoting alignment of parameter level sets by being minimal when gradients align. Let us introduce the matrix function $\mathbf{G} : \Omega \to \mathbb{R}^{d\times 2}$, with $\Omega \subset \mathbb{R}^d$ the physical domain, defined by

$$\mathbf{G}(x) \coloneqq \left[\nabla m_1 | \nabla m_2\right] = \begin{bmatrix} \partial_{x_1} m_1 & \partial_{x_1} m_2 \\ & \vdots \\ \partial_{x_d} m_1 & \partial_{x_d} m_2 \end{bmatrix}.$$

The gradients ∇m_1 and ∇m_2 are aligned at $x \in \Omega$ if the columns of $\mathbf{G}(x)$ are a multiple of each other, in which case the rank of $\mathbf{G}(x)$ is 1. One could seek to promote gradient alignment by minimizing the quantity $\int_\Omega \operatorname{rank}(\mathbf{G}(x))\, dx$. In practice, minimization of the rank of a matrix is notoriously difficult. However, the nuclear norm of a matrix, denoted $\|\cdot\|_*$, is often a good proxy for the rank [34]. The nuclear norm of a matrix is defined as the ℓ_1-norm of its singular values. We therefore define the nuclear norm joint regularization as

$$\mathcal{R}_*(m_1, m_2) \coloneqq \gamma \int_\Omega \|\mathbf{G}(x)\|_*\, dx, \tag{3.13}$$

with $\gamma > 0$.

3.3.1 Gradient of the nuclear norm joint regularization

We now show how to compute derivatives of (3.13), using the chain rule. Let us introduce the notation $f(\mathbf{G}) \coloneqq \|\mathbf{G}\|_*$, such that $\mathcal{R}_*(m_1, m_2) = \gamma \int_\Omega f(\mathbf{G}(x))\, dx$. Denoting the gradient of f with respect to the entries of matrix $\mathbf{G}$ by $\nabla f(\mathbf{G}) \in \mathbb{R}^{d\times 2}$, the first directional derivatives of (3.13) with respect to the inversion parameters m_i, $i = 1, 2$, are given by

$$\partial_{m_i} \mathcal{R}_*(m_1, m_2)\tilde{m}_i = \gamma \int_\Omega (\nabla f(\mathbf{G}), \partial_{m_i} \mathbf{G}(x)\tilde{m}_i)\, dx, \tag{3.14}$$

where

$$\partial_{m_1}\mathbf{G}(x)\tilde{m}_1 = \begin{bmatrix} \partial_{x_1}\tilde{m}_1 & 0 \\ \vdots & \\ \partial_{x_d}\tilde{m}_1 & 0 \end{bmatrix} \text{ and } \partial_{m_2}\mathbf{G}(x)\tilde{m}_2 = \begin{bmatrix} 0 & \partial_{x_1}\tilde{m}_2 \\ \vdots & \\ 0 & \partial_{x_d}\tilde{m}_2 \end{bmatrix},$$

and the inner-product for matrices $M = (m_{ij})_{ij}, N = (n_{ij})_{ij} \in \mathbb{R}^{d\times 2}$ is defined as $(M, N) = \sum_{i=1}^{d}\sum_{j=1}^{2} m_{ij}n_{ij}$. We next compute the gradient of the nuclear norm ∇f.

Given a full-rank matrix $M \in \mathbb{R}^{a\times b}$, i.e., $r \coloneqq \operatorname{rank}(M) = \min(a, b)$, and singular values $\{\sigma_k\}_{k=1}^{r}$, we define its (reduced) singular value decomposition (SVD) by $M = U\Sigma V^T$, with $U \in R^{a\times r}$, $V \in \mathbb{R}^{b\times r}$, and $\Sigma \in \mathbb{R}^{r\times r}$ a diagonal matrix containing the singular values of M, i.e., $\Sigma_{kk} = \sigma_k > 0$, $k = 1, \ldots, r$. The (i, j)-entry of the gradient of the nuclear norm is given by

$$(\nabla f(M))_{ij} = \sum_{k=1}^{r} \frac{\partial \sigma_k}{\partial m_{ij}} = \sum_{k=1}^{r} u_{ik}v_{jk},$$

where the second equality uses the singular value sensitivity [68]. The gradient of the nuclear norm with respect to the entries of M is then given by

$$\nabla f(M) = UV^T.$$

3.3.2 Modified nuclear norm joint regularization

The nuclear norm $f(M)$ is non-differentiable when the matrix M is not full-rank, corresponding to the case where at least one of the singular values vanishes. A workaround, similar to the one employed for the TV functional, is to define the modified nuclear norm

$$f_\varepsilon(M) \coloneqq \|M\|_{*,\varepsilon} = \sum_{k=1}^{\min(a,b)} \sqrt{\sigma_k^2 + \varepsilon}, \tag{3.15}$$

where $\varepsilon > 0$. We then define the modified nuclear norm joint regularization as

$$\mathcal{R}_{*,\varepsilon}(m_1, m_2) \coloneqq \gamma \int_\Omega f_\varepsilon(\mathbf{G}(x))\, dx, \tag{3.16}$$

with $\gamma > 0$.

The (i, j)-entry of the gradient of the modified nuclear norm (3.15) is given by

$$(\nabla f_\varepsilon(M))_{ij} = \frac{\partial}{\partial m_{ij}} \sum_{k=1}^{\min(a,b)} \sqrt{\sigma_k^2 + \varepsilon} = \sum_{k=1}^{r} \frac{\sigma_k}{\sqrt{\sigma_k^2 + \varepsilon}} \frac{\partial \sigma_k}{\partial m_{ij}},$$

where in the last expression the sum is up to r since, by definition of the rank of a matrix, $\sigma_k = 0$ for all $k > r$. Let us now introduce the diagonal matrix $W_\varepsilon \in \mathbb{R}^{r \times r}$, with entries $(W_\varepsilon)_{ii} = \sigma_i / \sqrt{\sigma_i^2 + \varepsilon}$. Using the expression for the sensitivity of the singular values [68], the gradient of the modified nuclear norm is then given by

$$\nabla f_\varepsilon(M) = U W_\varepsilon V^T. \tag{3.17}$$

The first directional derivatives of (3.16) with respect to the inversion parameters m_i, $i = 1, 2$, are given by

$$\partial_{m_i} \mathcal{R}_{*,\varepsilon}(m_1, m_2)\tilde{m}_i = \gamma \int_\Omega (\nabla f_\varepsilon(\mathbf{G}), \partial_{m_i} \mathbf{G}(x)\tilde{m}_i)\, dx, \tag{3.18}$$

The modified nuclear norm (3.15) is however not twice differentiable when two singular values are equal (crossing singular values). This is because the second-derivative requires the sensitivity of the individual singular vectors, which are not differentiable where singular values cross. We have not found a practical workaround for that singularity, and thus proceed with a gradient-based method to solve joint inverse problems regularized with the nuclear norm joint regularization; the solver is detailed in section 3.3.4.

In the rest of this chapter, we refer to the modified nuclear norm joint regularization (3.16) simply as the nuclear norm joint regularization.

3.3.3 Joint regularization, or structural similarity term?

An important question is whether the functional (3.16) requires to be supplemented by an actual regularization as in the case of the cross-gradient (section 3.1), or if it acts as a joint regularization as in the case of VTV (section 3.2). It turns out functional (3.16) acts like a joint regularization, as we verify numerically in section 3.4. However, we supply analytical evidence of that fact, in the case of two inversion parameters m_1 and m_2 defined over a domain $\Omega \subset \mathbb{R}^2$. In that case, $\mathbf{G}$ is a 2×2 matrix and we can derive an analytical expression for (3.16). As we show in the next paragraph, this expression displays strong resemblance with the VTV regularization.

For clarity of the presentation, we derive the analytical expression of (3.16) in the case $\varepsilon \equiv 0$ and $\gamma \equiv 1$. Using the analytical solution for the singular values of a matrix $M \in \mathbb{R}^{2\times 2}$ (see appendix (B.1)), we obtain

$$\begin{aligned}\int_\Omega \|\mathbf{G}(x)\|_* \, dx &= \int_\Omega \Big[\sigma_1(\mathbf{G}(x)) + \sigma_2(\mathbf{G}(x))\Big] \, dx \\ &= \frac{1}{\sqrt{2}} \int_\Omega \Big[\\ &\sqrt{|\nabla m_1|^2 + |\nabla m_2|^2 + \sqrt{(|\nabla m_1|^2 - |\nabla m_2|^2)^2 + 4(\nabla m_1 \cdot \nabla m_2)^2}} \\ &\quad + \sqrt{|\nabla m_1|^2 + |\nabla m_2|^2 - \sqrt{(|\nabla m_1|^2 - |\nabla m_2|^2)^2 + 4(\nabla m_1 \cdot \nabla m_2)^2}}\Big] \, dx.\end{aligned}$$

This expression simplifies in the following situations.

- if $\nabla m_1 \cdot \nabla m_2 = 0$, i.e., if both parameters are orthogonal to each other, the nuclear norm joint regularization is equal to the sum of two indepen-

dent TV regularizations,

$$\begin{aligned}\mathcal{R}_*(m_1,m_2) &= \frac{1}{\sqrt{2}}\int_\Omega \sqrt{|\nabla m_1|^2+|\nabla m_2|^2+||\nabla m_1|^2-|\nabla m_2|^2|}+ \\ &\qquad \sqrt{|\nabla m_1|^2+|\nabla m_2|^2-||\nabla m_1|^2-|\nabla m_2|^2|}\,dx \\ &= \int_\Omega |\nabla m_1|+|\nabla m_2|\,dx \\ &= \mathcal{R}_{\text{TV}}(m_1)+\mathcal{R}_{\text{TV}}(m_2).\end{aligned}$$

- if $\nabla m_1 \cdot \nabla m_2 = \pm|\nabla m_1|\,|\nabla m_2|$, i.e., if the level sets of both parameters are aligned, we recover the VTV functional,

$$\begin{aligned}\mathcal{R}_*(m_1,m_2) &= \frac{1}{\sqrt{2}}\int_\Omega \Big[\sqrt{|\nabla m_1|^2+|\nabla m_2|^2+\sqrt{(|\nabla m_1|^2+|\nabla m_2|^2)^2}} \\ &\qquad +\sqrt{|\nabla m_1|^2+|\nabla m_2|^2-\sqrt{(|\nabla m_1|^2+|\nabla m_2|^2)^2}}\Big]\,dx \\ &= \int_\Omega \sqrt{|\nabla m_1|^2+|\nabla m_2|^2}\,dx \\ &= \mathcal{R}_{\text{VTV}}(m_1,m_2).\end{aligned}$$

3.3.4 Numerical solution of joint inverse problems regularized with the nuclear norm joint regularization

To solve joint inverse problems regularized with the nuclear norm joint regularization, we use a BFGS quasi-Newton method with damped update [67]. That is, at each step k we compute the search direction p_k by computing $p_k = -H_k g_k$, where g_k is the gradient of the objective function to be minimized at step k, and H_k is a positive definite approximation to the inverse of the Hessian of that objective function. This approximation is improved by a rank-2 update at each step, using the formula

$$H_{k+1} = (I-\rho_k r_k y_k^T)H_k(I-\rho_k y_k r_k^T)+\rho_k r_k r_k^T, \tag{3.19}$$

where y_k is the difference between the gradient at step $k+1$ and k, $\rho_k \coloneqq 1/y_k^T r_k$, and r_k is the damped form of s_k, the difference between the parameter at step $k+1$ and k, and is defined as $r_k \coloneqq \theta_k s_k + (1-\theta_k) H_k y_k$, with

$$\theta_k \coloneqq \begin{cases} 1, & \text{if } s_k^T y_k \geq \alpha y_k^T H_k y_k, \\ \dfrac{(1-\alpha) y_k^T H_k y_k}{y_k^T H_k y_k - s_k^T y_k}, & \text{otherwise.} \end{cases}$$

The classic BFGS method requires the curvature condition, i.e., $s_k^T y_k > 0$, to be satisfied at all steps. The curvature condition is necessary to maintain positive definiteness of the BFGS approximation H_k. However, the curvature condition is only guaranteed to be true at all steps when the objective function is strictly convex everywhere, which is typically not the case for inverse problems. This can be remedied by performing an inexact line search verifying the Wolfe conditions [67]; but such a line search requires the computation of the gradient and can be computationally expensive (see section 1.3.2). Using a damped update allows us to apply a backtracking line-search, while avoiding to skip some updates of H_k entirely. In the numerical applications, we found that $\alpha = 0.2$ worked well. The BFGS formula (3.19) requires the initialization of the approximation of the inverse Hessian, i.e., the definition of H_0. BFGS-type methods perform well when the true Hessian is a compact operator [40, 41]. This is typically not the case for PDE-constrained inverse problems where the regularization is a non-compact differential operator. As a remedy, we set H_0 to the inverse of the Hessian of the regularization operator. This quantity is not available for the nuclear norm joint regularization, but based on the insight gained in section 3.3.3, we set H_0 to the inverse of the Hessian of the VTV joint regularization at the current parameter location.

3.4 Numerical examples

In this section, we present a comprehensive numerical comparison of the four joint regularization terms introduced in sections 3.1 to 3.3, i.e., the regularization (3.3) for the cross-gradient, the regularization (3.8) for the normalized cross-gradient, the vectorial total variation (3.10), and the nuclear norm regularization (3.15). Reconstructions obtained with these joint regularization terms are compared with each other, and with the reconstructions obtained from solving a joint inverse problem with two independent TV regularizations. The parameters for all joint regularization terms are selected empirically as leading to the best reconstructions. The values of ε are chosen small enough to provide reconstructions with sharp interfaces, but large enough to avoid unreasonable numerical difficulties (see for instance the discussion in [6]).

The four joint regularization terms are compared in three examples covering both types of joint inverse problems (1.35) and (1.36). In section 3.4.1, we combine two uncoupled Poisson inverse problems to create a joint inverse problem (1.36). To justify the use of a joint inversion, we assume to have knowledge about the truth parameter fields of both Poisson problems having similar structure. In section 3.4.2, we compare how the use of joint regularization terms can help improve the reconstruction of the bulk modulus and the density in an acoustic wave equation, an example of joint inverse problem (1.35). In section 3.4.3, we formulate a multi-physics joint inverse problem (1.36), combining a Poisson inverse problem and an acoustic wave inverse problem. The Poisson parameter and the wave velocity are assumed to have similar structure.

In all examples, the domain is a 2D unit square, meshed by regular isosceles square triangles obtained by cutting in half $N \times N$ squares; we define

the mesh size parameter $h \coloneqq 1/N$. All data are generated synthetically at the truth parameter fields, then polluted by adding independent and identically distributed Gaussian noise; the noise level is specific to each example. We use continuous finite elements to discretize the inverse problems and the PDE constraints, with the state, adjoint, incremental state, and incremental adjoint variables using quadratic elements, and the parameter fields using linear elements. For all examples, the solvers are implemented in Python with the finite-element library FEniCS [61, 62]. For the examples in section 3.4.1 and 3.4.3, we used the optimization routines from hIPPYlib [84], a Python library for deterministic and Bayesian inverse problems.

3.4.1 Joint Poisson inverse problems

In the first two examples, we solve a joint inverse problem (1.36) with two parameters, m_1 and m_2. Considered separately, each parameter m_i, $i = 1, 2$, is solution to a Poisson inverse problem,

$$\begin{gathered} m_i \coloneqq \arg\min_m \left\{ \frac{1}{2}|B_i u - \mathbf{d}_i|^2 + \gamma_i \int_\Omega \sqrt{|\nabla m|^2 + \varepsilon}\, dx \right\}, \\ \text{where,} \\ \begin{cases} -\nabla\cdot(e^m \nabla u) = 1, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases} \end{gathered} \tag{3.20}$$

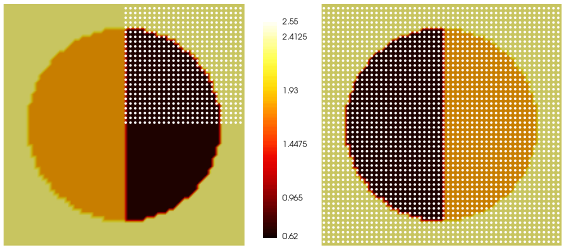
The operators B_i represent pointwise observation operators, and the data $\mathbf{d}_i$ are synthetic observations polluted with 2% Gaussian noise. The domain Ω is meshed with 8192 identical isosceles square triangles ($h = 1/64$).

The differences between the inverse problems for m_1 and m_2 reside in the truth parameter fields, and in the number and locations of the observation points. In the first example (section 3.4.1.1), the truth parameter fields

differ but have interfaces at the same spatial locations. In the second example (section 3.4.1.2), some interfaces in the truth parameter field for m_2 do not appear in the truth parameter field for m_1. In both examples, the data for parameter m_1 are observations that only cover the top-right quadrant of the domain, whereas the data for parameter m_2 are observations distributed over the entire domain. The locations of the observations for each inverse problem are shown in figures 3.4 and 3.7.

3.4.1.1 Truth parameter fields have identical interface locations

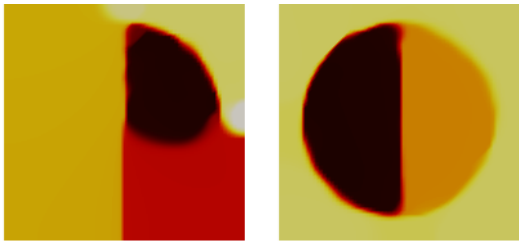
In the first example, the truth parameter fields for both m_1 and m_2 have sharp interfaces at the same locations (see figure 3.4). For reference,



(a) m_1 (b) m_2

Figure 3.4: Truth parameter fields for (a) m_1 and (b) m_2 in the example of section 3.4.1.1. White dots in (a) and (b) indicate the location of the pointwise observations. The observations for parameter m_1 only cover the top-right quadrant of the domain, forming a square lattice of 25×25 pointwise observations. The observations for parameter m_2 form a square lattice of 50×50 pointwise observation distributed over the entire domain.

we show in figure 3.5 the reconstructions for parameters m_1 and m_2 obtained by solving two independent inverse problems (3.20). The reconstructions for all four joint inverse problems are shown in figure 3.6, and the corresponding values of the relative medium misfit are given in table B.1.



(a) m_1 (b) m_2

Figure 3.5: Reconstructions for the parameter fields (a) m_1 and (b) m_2, obtained by solving the inverse problem (3.20) regularized with TV ($\varepsilon = 10^{-3}$, $\gamma_1 = 3 \cdot 10^{-7}$, and $\gamma_2 = 4 \cdot 10^{-7}$). The legend for both plots is as in figure 3.4.

The reconstructions for parameter m_2 do not vary significantly from one another (figure 3.6b); due to the large number of observation points, this parameter was already well reconstructed by an independent inverse problem (figure 3.5b). We observe an improvement in the reconstruction of parameter m_1 for all four joint inverse problems, over the independent reconstruction shown in figure 3.5a. Using the cross-gradient only marginally improves the reconstruction for parameter m_1, most likely because the independent reconstruction for m_1 shows large areas of constant values, where the cross-gradient term is equal to zero; these areas can therefore not be improved by the cross-gradient. The normalized cross-gradient improves over the cross-gradient but fails to fully recover the circular interface. Both the VTV joint regularization and the nuclear norm joint regularization perform better in this example, and lead to reconstructions that contain all sharp interfaces in the target image.

3.4.1.2 Truth parameter fields have different interface locations

Here, the only difference with the previous example is that the truth parameter field for m_1 no longer has a vertical discontinuity along the line $x = 0.5$

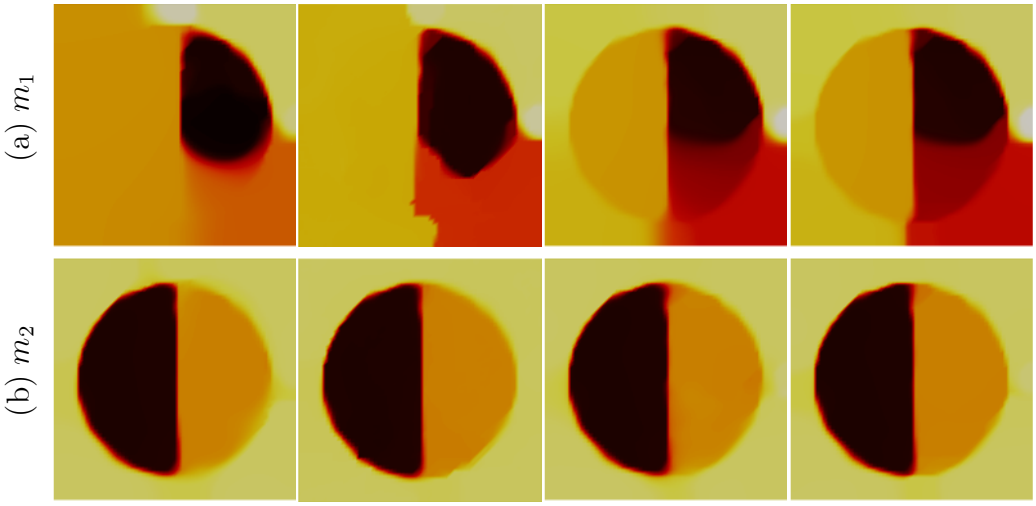
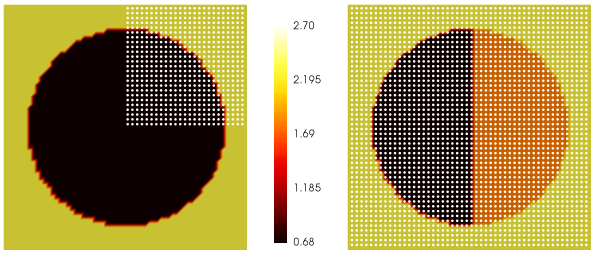


Figure 3.6: Reconstructions for the parameter fields (a) m_1 and (b) m_2, obtained by solving a joint inverse problem (1.36) regularized with (i) the cross-gradient ($\gamma = 2 \cdot 10^{-8}$) paired with two independent TV regularizations, (ii) the normalized cross-gradient ($\gamma = 6 \cdot 10^{-6}$ and $\varepsilon = 10^{-3}$) paired with two independent TV regularizations, (iii) the VTV joint regularization ($\gamma = 3 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$), and (iv) the nuclear norm joint regularization ($\gamma = 3 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$). The parameters for the independent TV regularizations are as for the independent inverse problems (see caption of figure 3.5). The legend is as in figure 3.4.

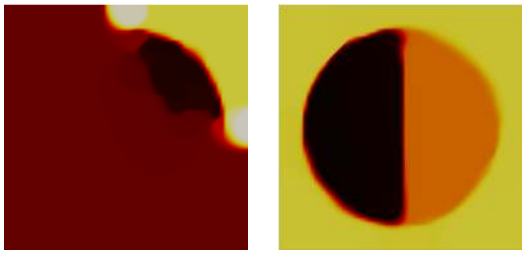
(see figure 3.7). For reference, in figure 3.8, we show the reconstructions for parameters m_1 and m_2 obtained by solving two independent inverse problems (3.20). The reconstructions for the four joint inverse problems are shown in figure 3.9, and the corresponding values of the relative medium misfit are given in table B.2.

For parameter m_2, we are in the situation of the first example (section 3.4.1.1), and the reconstructions from the four joint inverse problems do not vary significantly from one another (see figure 3.9b). However, we observe



(a) m_1 (b) m_2

Figure 3.7: Truth parameter fields for (a) m_1 and (b) m_2 in the example of section 3.4.1.2. White dots in (a) and (b) indicate the location of the pointwise observations (see caption of figure 3.4).



(a) m_1 (b) m_2

Figure 3.8: Reconstructions for the parameter fields (a) m_1 and (b) m_2, obtained by solving the inverse problem (3.20) regularized with TV ($\varepsilon = 10^{-3}$, $\gamma_1 = 4 \cdot 10^{-7}$, and $\gamma_2 = 4 \cdot 10^{-7}$). The legend for both plots is the same as in figure 3.7.

differences among the reconstructions for parameter m_1. The joint inverse problem using the cross-gradient only marginally improves the reconstruction for parameter m_1. Here, the use of the normalized cross-gradient does not show any improvement over the cross-gradient. As in the first example, both the VTV joint regularization and the nuclear norm joint regularization perform the best, and their corresponding reconstructions contain all sharp interfaces present in the target image. But we also see, in figures 3.9a (iii) and (iv), the presence of a vertical discontinuity that was not in the target image 3.8a.

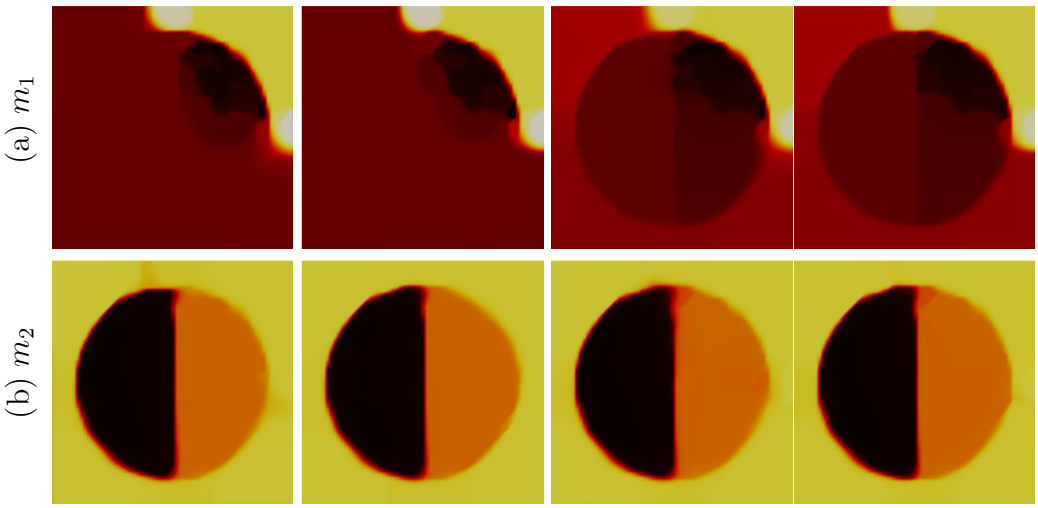


Figure 3.9: Reconstructions for the parameter fields (a) m_1 and (b) m_2, obtained by solving a joint inverse problem (1.36) regularized with (i) the cross-gradient paired with 2 independent TV regularizations ($\gamma = 5 \cdot 10^{-9}$), (ii) the normalized cross-gradient paired with 2 independent TV regularizations ($\gamma = 7 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$), (iii) the VTV joint regularization ($\gamma = 4 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$), and (iv) the nuclear norm joint regularization ($\gamma = 4 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$). The parameters for the independent TV regularizations are the ones selected for the independent inverse problems (see caption figure 3.8). The legend for all plots is the same as in figure 3.7.

This ghost interface is due to the presence of such a discontinuity in the other parameter m_2, and highlights the tendency of the VTV joint regularization and nuclear norm joint regularization to super-impose discontinuities in both parameters. Note, however, that the amplitude of this ghost interface is small compared to the amplitudes of the other true interfaces.

3.4.2 Joint inversion of bulk modulus and density in the acoustic wave equation

We now study an example of joint inverse problem (1.35), where both parameters enter the same equation, namely the acoustic wave equation.

3.4.2.1 Problem description

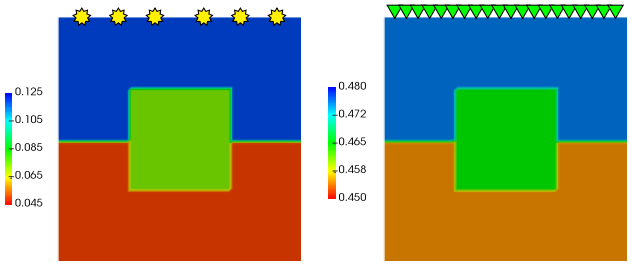
The propagation of acoustic waves depends on the bulk modulus λ and the density ρ of the medium of propagation. The acoustic wave equation (1.25) was introduced in section 1.4.1. In this section, we assume that both the bulk modulus λ and the density ρ are unknown, and we use the acoustic wave inverse problem (1.28) defined in section 1.4.2.

3.4.2.2 Solution of the acoustic wave joint inverse problem

Because of the coupling between parameters α and β coming from the acoustic wave equation, the inverse problem (1.28) could be regularized by two independent TV regularizations, i.e., $\mathcal{R}(\alpha, \beta) = \mathcal{R}_{\mathrm{TV},\varepsilon}(\alpha) + \mathcal{R}_{\mathrm{TV},\varepsilon}(\beta)$ [32]. However, the acoustic wave inverse problem with independent regularizations can be difficult to solve. Going beyond the use of ad-hoc methods to handle both parameters at once, some researchers have looked at solving (1.28) as a joint inverse problem [60, 63]. Previous attempts have not considered the normalized cross-gradient, the VTV regularization, or the nuclear norm regularization. In this section, we study how the use of joint regularization can result in improved reconstructions for α and β.

In our numerical tests, we use 6 sources, $f_i(\mathbf{x}, t)$, located on the top boundary of the domain at 0.1, 0.25, 0.4, 0.6, 0.75, and 0.9 from the left boundary (yellow stars in figure 3.10a); each source is a point source in space,

and a Ricker wavelet in time with a peak frequency of 2Hz. The data are recorded at 20 locations equally spaced along the top boundary (green triangles in figure 3.10b), and polluted by independent Gaussian noise with zero mean and variance corresponding to a signal-to-noise ratio of 20 dB. The boundary conditions are homogeneous Neumann boundary conditions along the top boundary $\partial\Omega_n = [0,1] \times \{1\}$, and absorbing boundary conditions along the left, bottom, and right boundaries $\partial\Omega_a = \{0,1\} \times [0,1] \cup [0,1] \times \{0\}$. The truth parameter fields for α and β are shown in figure 3.10; they correspond to an acoustic wave velocity varying from 2km/s to 3km/s[1]. Those are typical values for the surface of the earth (see for instance [64, 85]). The domain is meshed with 800 identical isosceles square triangles ($h = 1/20$).



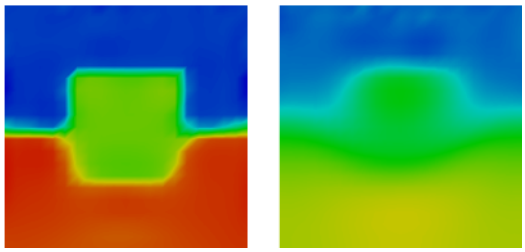
(a) α (b) β

Figure 3.10: Truth parameter fields for (a) α and (b) β in the joint acoustic inverse problem (1.28) used in section 3.4.2. The yellow stars in (a) indicate the locations of the sources. The green triangles in (b) indicate the locations of the pointwise observations.

For reference, we show in figure 3.11 the reconstructions for parameters α and β obtained by solving (1.28) regularized with two independent TV regularizations. Whereas parameter α is well reconstructed, the reconstruc-

[1]with the following units: distance in km, velocity in km/s, density in g/cm^3, and bulk modulus in GPa.

tion for parameter β appears rather blurry. We next solve (1.28) regularized



(a) α (b) β

Figure 3.11: Reconstructions for the parameter fields (a) α and (b) β when solving (1.28) regularized with two independent TV regularizations ($\varepsilon = 10^{-3}$, $\gamma_\alpha = 5 \cdot 10^{-6}$, and $\gamma_\beta = 9 \cdot 10^{-6}$).

with the four joint regularization terms studied. The results are shown in figure 3.12, and the corresponding values of the relative medium misfit are given in table B.3.

The different reconstructions for parameter α (figure 3.12a) do not differ significantly from each other, as α is also well reconstructed by solving the inverse problem with two independent regularizations (figure 3.11a). However, the use of a joint regularization improves the quality of the reconstruction for the parameter β (figure 3.12b). Whereas the use of the cross-gradient only results in marginal improvement to the reconstruction in figure 3.11b, the use of the normalized cross-gradient allows to recover all interfaces more clearly. The best reconstructions are obtained using the VTV or the nuclear norm joint regularizations.

3.4.3 Multi-physics inverse problem

As last problem, we study a multi-physics inverse problem (1.36) combining a Poisson inverse problem and an acoustic wave inverse problem (as-

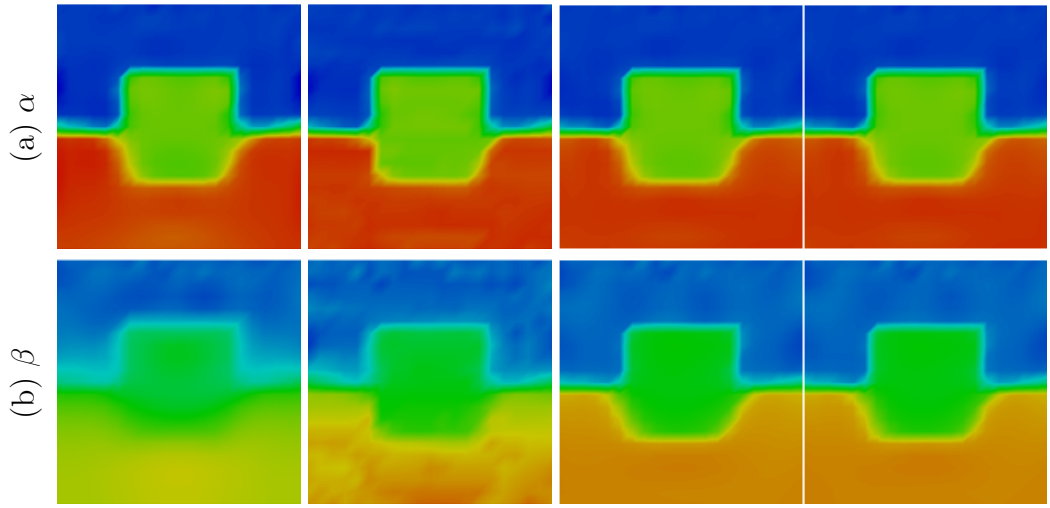


Figure 3.12: Reconstructions for the parameter fields (a) α and (b) β, obtained by solving inverse problem (1.28) regularized with (i) the cross-gradient ($\gamma = 10^{-2}$) paired with two independent TV regularizations, (ii) the normalized cross-gradient ($\gamma = 9\cdot 10^{-6}$ and $\varepsilon = 10^{-6}$) paired with two independent TV regularizations, (iii) the VTV joint regularization ($\gamma = 7\cdot 10^{-6}$ and $\varepsilon = 10^{-3}$), and (iv) the nuclear norm joint regularization ($\gamma = 7\cdot 10^{-6}$ and $\varepsilon = 10^{-3}$). The parameters for the independent TV regularizations are the ones selected for the independent inverse problems (see caption in figure 3.11). The legend for all plots is the same as in figure 3.10.

suming the density ρ is known). The Poisson inverse problem is identical to the one used in section 3.4.1,

$$\min_m \left\{ \frac{1}{2}|Bu - \mathbf{d}|^2 + \gamma_m \int_\Omega \sqrt{|\nabla m|^2 + \varepsilon}\, dx \right\}, \quad \text{where,} \quad \begin{cases} -\nabla\cdot(e^m \nabla u) = 1, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases} \tag{3.21}$$

The observations form a square lattice of 20×20 points equally distributed over the entire domain (white dots in figure 3.13a). The data are polluted with

1% Gaussian noise. For the acoustic wave inverse problem, we set parameter β to $\beta \equiv 1$ in (1.28), and invert only for the parameter $\alpha = 1/\lambda = 1/c^2$,

$$\min_{\alpha} \left\{ \frac{1}{2} \int_0^T |Bu(t) - \mathbf{d}(t)|^2 \, dt + \gamma_\alpha \int_\Omega \sqrt{|\nabla \alpha|^2 + \varepsilon} \right\} dx,$$

where,

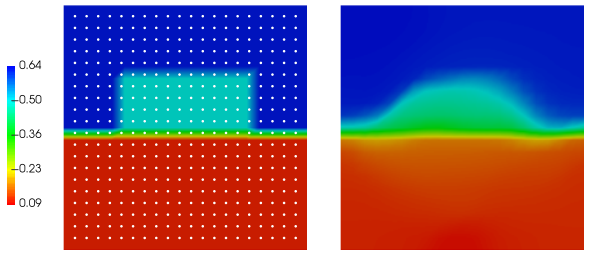
$$\begin{aligned} \alpha \ddot{u} - \Delta u &= f_\alpha, \quad \text{in } \Omega \times (0,T), \\ u(\mathbf{x},0) = \dot{u}(\mathbf{x},0) &= 0, \quad \text{in } \Omega, \\ \begin{cases} \nabla u \cdot \mathbf{n} = 0, & \text{on } \partial\Omega_n \times (0,T), \\ \nabla u \cdot \mathbf{n} = -\sqrt{\alpha}\dot{u}, & \text{on } \partial\Omega_a \times (0,T). \end{cases} \end{aligned} \tag{3.22}$$

We use a single source f_α of frequency 4Hz, located at $(0.5, 0.1)$ (yellow star in figure 3.14a), and 20 pointwise observations equally spaced along the top boundary (green triangles in figure 3.14a). The boundary conditions, the noise level in the data, the mesh, and the numerical discretization are as in section 3.4.2.

Although the goal of this example is to study how each joint regularization performs in a multi-physics joint inverse problem (1.36), we mention a potential application to seismic imaging for this specific combination of inverse problems. As discussed in section 1.5.3, a successful full-waveform inversion requires low-frequency data, which are rarely available in practice, or a good starting model. Practitioners have traditionally built these starting models using other, more rudimentary imaging techniques [9]. One potential application of joint inverse problems is to allow an auxiliary physical phenomenon, e.g., electromagnetic waves, to provide the missing low-frequency information for full-waveform inversion. When taking the frequency of the electromagnetic source term toward the low-frequency regime, Maxwell's equation behaves more and more like an elliptic PDE [42]; this combination of elliptic inverse

problem and acoustic wave inverse problem corresponds to the example of this section.

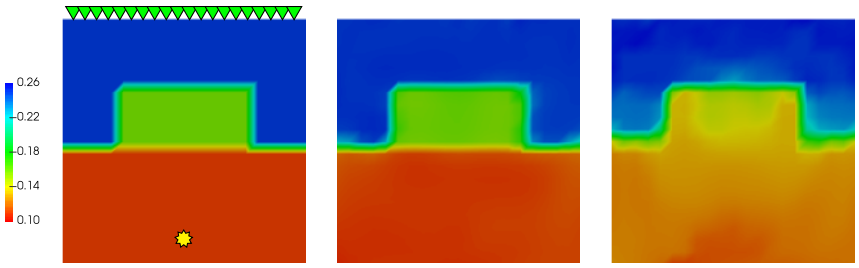
As a reference, we first show the reconstructions for parameters m and α when (3.21) and (3.22) are solved independently. In figure 3.13b, we show the reconstruction for the Poisson inverse problem (3.22), where the horizontal interface is well reconstructed, but the shape of the rectangular perturbation is smeared out. For the acoustic wave inverse problem (3.22), we show two



(a) truth (b) reconstruction

Figure 3.13: Plots of (a) truth parameter field for m in the Poisson inverse problem (3.21), and (b) its reconstruction using TV regularization ($\gamma_m = 2 \cdot 10^{-8}$ and $\varepsilon = 10^{-3}$), and initial parameter field set to a constant value of 0.625. The white dots in (a) indicate the location of the pointwise observations.

reconstructions in figure 3.14, one with a source f_α of frequency 2Hz (figure 3.14b), and one with a source f_α of frequency 4Hz (figure 3.14c). Whereas the reconstruction at 2Hz is excellent, the reconstruction at 4Hz lacks sufficient low-frequency information and converges toward a local minimum, missing the horizontal discontinuity present in the truth parameter field (figure 3.14a). The reconstructions for all four joint inverse problems, with a seismic source f_α of frequency 4Hz, are shown in figure 3.15, and the corresponding values of the relative medium misfit are given in table B.4.



(a) truth (b) reconstr. 2Hz (c) reconstr. 4Hz

Figure 3.14: Plots of (a) truth parameter field for α in the acoustic inverse problem (3.22), and (b,c) its reconstructions using TV regularization ($\gamma_\alpha = 3 \cdot 10^{-8}$ and $\varepsilon = 10^{-3}$), and initial value for the parameter field set to 0.25, and a source f_α of frequency (b) 2Hz, and (c) 4Hz. The green triangles in (a) indicate the locations of the pointwise observations, and the yellow star in (a) indicates the location of the source.

The use of either the cross-gradient term, or the normalized cross-gradient term, improves the reconstruction for the Poisson parameter m (figures 3.15a, (i) and (ii)), compared to the reconstruction from the Poisson inverse problem (3.21) alone (figure 3.13b). However, neither of the cross-gradient terms brings any improvement to the reconstruction of the acoustic wave velocity (figures 3.15b, (i) and (ii)); in particular, the reconstructions do not show the horizontal discontinuity that was missing in the reconstruction of the acoustic wave velocity alone (figure 3.14c). On the other hand, the use of either the VTV joint regularization, or the nuclear norm joint regularization, leads to significant improvements in the reconstruction of the acoustic wave velocity (figures 3.15b, (iii) and (iv)). Both reconstructions contain all features of the truth parameter field (figure 3.14a); most noticeably, the horizontal discontinuity that was missing in the independent reconstruction (figure 3.14c) is now fully reconstructed. The use of the VTV joint regularization provides only marginal improvement to the reconstruction of the Poisson parameter m,

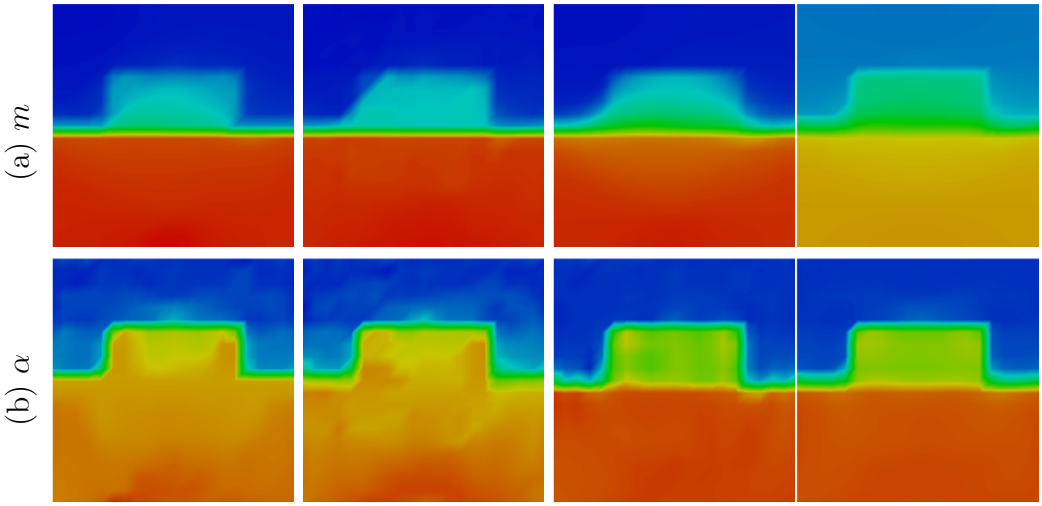


Figure 3.15: Reconstructions for the parameter fields (a) m in (3.21) and (b) α in (3.22), obtained by solving a joint inverse problem with seismic source f_α of frequency 4Hz, and regularized with (i) the cross-gradient ($\gamma = 8\cdot 10^{-7}$) paired with two TV regularizations, (ii) the normalized cross-gradient ($\gamma = 8\cdot 10^{-8}$ and $\varepsilon = 10^{-5}$) paired with two TV regularizations, (iii) the VTV joint regularization ($\gamma = 4\cdot 10^{-8}$ and $\varepsilon = 10^{-3}$), and (iv) the nuclear norm joint regularization ($\gamma = 5\cdot 10^{-7}$ and $\varepsilon = 10^{-3}$). The parameters for the independent TV regularizations are as for the independent inverse problems (see captions of figures 3.13 and 3.14). Legend is the same as in figures 3.13 and 3.14.

in terms of relative medium misfit (table B.4); however, the shape of the rectangular perturbation, which was smeared out in the reconstruction from the Poisson inverse problem alone (figure 3.13b), is much more clear in figure 3.15a(iii). The reconstruction of the Poisson parameter obtained with the nuclear norm joint regularization indicates that the optimization converged to a local minimum. Despite all discontinuities present in the truth parameter field (figure 3.13a) being clearly reconstructed in figure 3.15a(iv), the values of the parameters are significantly different. Similar, or worse, performance was observed when setting H_0 to be a multiple of the identity matrix in the BFGS

solver [67]. Moreover, almost identical results were obtained when solving the Poisson-acoustic joint inverse problem, regularized by VTV, using the BFGS method described in section 3.3.4. We can therefore conjecture that the poor performance of the nuclear norm joint regularization, in the case of a multi-physics joint inverse problem, is due to the use of a gradient-based method for the solution of the joint inverse problem. The significant difference in the structure of the gradients, coming from the Poisson and acoustic wave inverse problems, dictate the use of a second-order method, in order to harmonize the search direction, and avoid diverging or falling into local minima.

3.5 Conclusion

We conducted a comprehensive review of regularization terms for joint inverse problems. We considered two types of joint inverse problems: (1) joint inverse problems combining multiple independent inverse problems, and (2) joint inverse problems where all inversion parameters depend on the same physics. Based on current literature review, we identified three joint regularization terms for this study that are amenable for the large-scale solution of PDE constrained joint inverse problems. The cross-gradient is a popular choice in geophysical applications and operates by forcing alignment of the level sets. The normalized cross-gradient was designed to overcome some of the potential weaknesses of the cross-gradient term. The vectorial total variation is an extension of the total variation regularization to joint inverse problems, and originated from the imaging community. In addition, we introduced a fourth novel joint regularization term based on the nuclear norm of a gradient matrix. The comparison of these joint regularization terms was performed over three examples: (1) a joint Poisson inverse problem for which the truth parameter

fields are known to share structures, (2) an acoustic wave inverse problem where we invert for the bulk modulus and the density, and (3) a joint Poisson-acoustic inverse problem providing an example of multi-physics joint inverse problem.

Based on this study, we recommend to use the vectorial total variation joint regularization. It led to excellent results for all examples, and scalable, efficient solvers for joint inverse problems regularized with vectorial total variation are available (see section 4). The cross-gradient similarity term does not improve significantly over independent reconstructions. In particular, it can fail to reconstruct some edges entirely, as the cross-gradient term is minimum where one parameter field is constant. The normalized cross-gradient similarity term leads to a joint inverse problem that is numerically challenging. Even though it improves over the cross-gradient, the improvement is generally minor, and the reconstructions did not compare favorably with the ones obtained with vectorial total variation. The nuclear norm joint regularization showed encouraging results, even leading to slightly better reconstructions than with vectorial total variation for some examples, but is missing a formulation that is twice differentiable for its use with Newton's method.

The last example showed evidence that a Poisson inverse problem and an acoustic wave inverse problem can be solved jointly as a way to provide missing low-frequency information to the acoustic wave inverse problem. Such results were however only made possible when regularizing with the vectorial total variation joint regularization.

Chapter 4

A primal-dual Newton method for the solution of joint inverse problems regularized with vectorial total variation

In chapter 3, we compared four joint regularizations over three examples of multi-parameter and multi-physics inverse problems, and identified vectorial total variation as the best regularization to reconstruct sharp edges. In section 1.3, we discussed the importance of using Newton-type methods to solve inverse problems constrained by PDEs. However, the use of the classical Newton linearization (section 1.3) on joint inverse problems regularized with VTV leads to extremely poor performance (see section 4.3); the number of PDE solves required to converge increases dramatically when the hyperparameter ε decreases, corresponding to reconstructions with increasingly sharper edges. Motivated by this, we developed a primal-dual Newton method whose performance depends only mildly on ε. The introduction of a dual variable allows an equivalent expression of the first-order optimality system that avoids the appearance of unknowns in both the numerator and the denominator. Compared to the primal system, the primal-dual first-order optimality system effectively reduces the nonlinearity of the problem as it involves only products of the unknowns. Through a comprehensive range of numerical examples, we demonstrate scalability of our solver with respect to the hyperparameter ε, the mesh size, and the number of parameter fields n that we invert for.

In section 4.1, we study vectorial total variation in the case of an arbitrary number of inversion parameters, and we analyze its action on the solution of a joint inverse problem. We introduce the primal-dual Newton solver for joint inverse problems regularized with VTV in section 4.2. We carefully discuss discretization issues (section 4.2.2), and the solution of the Newton system (section 4.2.3). The numerical section 4.3 contains three subsections where we demonstrate scalability of the primal-dual Newton solver with respect to the hyperparameter ε (section 4.3.1), the mesh size (section 4.3.2), and the number of inversion parameters (section 4.3.3). We provide some concluding remarks in section 4.4.

4.1 Vectorial total variation

As a preamble to the VTV joint regularization, we consider a joint regularization that allows reconstructions with sharp edges, but does not impose any coupling between the inversion parameters; we call it joint TV regularization (section 4.1.1). In section 4.1.2, we introduce the formulation of VTV with n parameter fields. In section 4.1.3, we compute derivatives of VTV, and carry out a comparative analysis of VTV and joint TV.

4.1.1 Joint total variation regularization

Let us assume that for all $i = 1, \dots, n$, $m_i \in W^{1,1}(\Omega)$. We define the joint TV regularization as the sum of n independent TV regularizations,

$$\mathcal{R}_{j\text{TV}}(m_1, \dots, m_n) \coloneqq \sum_{i=1}^{n} \mathcal{R}_{\text{TV},\varepsilon}(m_i), \tag{4.1}$$

where $\mathcal{R}_{\text{TV},\varepsilon}$ was defined in (1.6). The derivatives of a joint regularization provide information about its action onto the solution of the joint inverse problem.

In the case of the joint TV regularization (4.1), with hyperparameter $\varepsilon = 0$, the weak form of the gradient [26] evaluated at a point $\mathbf{m} \coloneqq (m_1, \ldots, m_n)$ and acting in a direction $\tilde{\mathbf{m}} \coloneqq (\tilde{m}_1, \ldots, \tilde{m}_n)$ is given, block-wise for any $i = 1, \ldots, n$, by

$$\delta_{m_i} \mathcal{R}_{j\mathrm{TV}}(\mathbf{m}; \tilde{\mathbf{m}}) = \int_\Omega \frac{\nabla m_i \cdot \nabla \tilde{m}_i}{|\nabla m_i|} dx. \tag{4.2}$$

Then the action of the Hessian of the joint TV regularization, at a point $\mathbf{m}$, and along a direction $\hat{\mathbf{m}} \coloneqq (\hat{m}_1, \ldots, \hat{m}_n)$ is given, block-wise for any $i, j = 1, \ldots, n$, by

$$\delta^2_{m_i, m_j} \mathcal{R}_{j\mathrm{TV}}(\mathbf{m}; \tilde{\mathbf{m}}, \hat{\mathbf{m}}) = \begin{cases} \displaystyle\int_\Omega \frac{\nabla \tilde{m}_i \cdot \nabla \hat{m}_i}{|\nabla m_i|} - \frac{(\nabla \tilde{m}_i \cdot \nabla m_i)(\nabla m_i \cdot \nabla \hat{m}_i)}{|\nabla m_i|^3} \, dx, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases} \tag{4.3}$$

The strong form of the Hessian $\mathcal{H}$ in (4.2) is an anisotropic diffusion operator, $\mathcal{H}\hat{\mathbf{m}} = -\nabla \cdot (A_{j\mathrm{TV}}(\mathbf{m}) \nabla \hat{\mathbf{m}})$. In the case of the joint TV, the diffusion tensor $A_{j\mathrm{TV}}(\mathbf{m})$ is block-diagonal with i^{th} diagonal entry given by

$$\left(A_{j\mathrm{TV}}(\mathbf{m})\right)_i = \frac{1}{|\nabla m_i|} \left(I - \frac{\nabla m_i \nabla m_i^T}{|\nabla m_i|^2} \right). \tag{4.4}$$

The action of the Hessian $\mathcal{H}$ can be analyzed through its diffusion tensor, which can be characterized by its eigen-decomposition. For any $i = 1, \ldots, n$, we have the following eigenpairs,

$$\left(\begin{bmatrix} \vdots \\ 0 \\ \nabla m_i \\ 0 \\ \vdots \end{bmatrix}, 0 \right), \left(\begin{bmatrix} \vdots \\ 0 \\ (\nabla m_i)^\perp \\ 0 \\ \vdots \end{bmatrix}, \frac{1}{|\nabla m_i|} \right).$$

The joint TV regularization acts on the search direction of each parameter m_i independently from the other parameters, and in the same way the TV functional would for a single inverse problem, i.e., by preserving sharp interfaces (large values of $|\nabla m_i|$) and smoothing out along orthogonal directions.

4.1.2 Formulation

The VTV joint regularization (see also section 3.2), in addition to preserving sharp discontinuities in the reconstructions, imposes some coupling between the parameter fields. Let us assume that for all $i = 1, \ldots, n$, $m_i \in W^{1,1}(\Omega)$. The VTV joint regularization is defined as

$$\mathcal{R}_{\text{VTV}}(\mathbf{m}) = \int_\Omega \sqrt{\sum_{i=1}^{n} |\nabla m_i|^2}\, dx. \tag{4.5}$$

From the equality $\sqrt{\sum_k a_k} \leq \sum_k \sqrt{a_k}$, we see that when $\varepsilon = 0$ in (4.1),

$$\mathcal{R}_{\text{VTV}}(\mathbf{m}) \leq \mathcal{R}_{j\text{TV}}(\mathbf{m}).$$

This shows that the VTV functional is well defined when all $m_i \in W^{1,1}(\Omega)$. However formulation (4.5) will be non-differentiable where $|\nabla m_i| = 0$ for all $i = 1, \ldots, n$. We therefore introduce the modified VTV defined as

$$\mathcal{R}_{\text{VTV}}(m) = \int_\Omega \sqrt{\sum_{i=1}^{n} |\nabla m_i|^2 + \varepsilon}\, dx, \tag{4.6}$$

with $\varepsilon > 0$.

4.1.3 Derivatives

Preliminary analysis of the VTV functional was already carried out in section 3.2. We here focus on gaining additional insight about the VTV

functional through careful study of its derivatives. We study the gradient of the VTV functional (4.5), and the action of its Hessian in a given direction. In weak form, the i^{th} block of the gradient is given, for any $i = 1, \ldots, n$, by

$$\delta_{m_i} \mathcal{R}(\mathbf{m}; \tilde{\mathbf{m}}) = \int_\Omega \frac{\nabla m_i \cdot \nabla \tilde{m}_i}{\sqrt{\sum_{k=1}^n |\nabla m_k|^2}} \, dx \tag{4.7}$$

This expression resembles the one for the independent TV regularization (4.2), with the difference that the denominator now is the root-mean square of the norms of the gradients of all parameter fields; we denote that quantity by $|\nabla \mathbf{m}| \coloneqq \sqrt{\sum_{k=1}^n |\nabla m_k|^2}$. The action of the Hessian of the VTV functional, at a point $\mathbf{m}$, along a direction $\hat{\mathbf{m}} = (\hat{m}_1, \ldots, \hat{m}_n)$, is given, block-wise, by

$$\begin{aligned} &\delta^2_{m_i, m_j} \mathcal{R}_{\text{VTV}}(\mathbf{m}; \tilde{\mathbf{m}}, \hat{\mathbf{m}}) = \\ &\begin{cases} \displaystyle\int_\Omega \frac{1}{|\nabla \mathbf{m}|} \left[\nabla \tilde{m}_i \cdot \nabla \hat{m}_i - \frac{(\nabla \tilde{m}_i \cdot \nabla m_i)(\nabla m_i \cdot \nabla \hat{m}_i)}{|\nabla \mathbf{m}|^2} \right] dx, & \text{if } i = j, \\ \displaystyle -\int_\Omega \frac{1}{|\nabla \mathbf{m}|^3} (\nabla \tilde{m}_i \cdot \nabla m_i)(\nabla m_j \cdot \nabla \hat{m}_j) \, dx, & \text{otherwise.} \end{cases} \end{aligned} \tag{4.8}$$

Then the (i, j)-block of the diffusive tensor, $A_{\text{VTV}}(\mathbf{m})$, of the strong form of the Hessian of the VTV functional is given by

$$(A_{\text{VTV}}(\mathbf{m}))_{i,j} = \frac{1}{|\nabla \mathbf{m}|} \left(\delta_{ij} I - \frac{\nabla m_i \nabla m_j^T}{|\nabla \mathbf{m}|^2} \right), \tag{4.9}$$

where δ_{ij} is 1 when $i = j$ and 0 otherwise.

We can gain additional insight about the difference of behaviour between the joint TV regularization and the VTV regularization by comparing the eigenpairs of their diffusion tensors. First of all, we notice that, unlike for the joint TV regularization, for any $i = 1, \ldots, n$, the direction $[\ldots, 0, \nabla m_i, 0, \ldots]^T$ is not in the kernel of the diffusion tensor A_{VTV}. However,

the following is in the kernel of A_{VTV},

$$A_{\text{VTV}}(\mathbf{m}) \begin{bmatrix} \nabla m_1 \\ \vdots \\ \vdots \\ \nabla m_n \end{bmatrix} = 0.$$

In the case of the VTV joint regularization, all directions ∇m_i have to be considered altogether, instead of independently as it was the case for the joint TV regularization. For the directions that are smoothed out, we see that for any $i = 1, \ldots, n$,

$$A_{\text{VTV}}(\mathbf{m}) \begin{bmatrix} \vdots \\ 0 \\ (\nabla m_i)^{\perp} \\ 0 \\ \vdots \end{bmatrix} = \frac{1}{|\nabla \mathbf{m}|} \begin{bmatrix} \vdots \\ 0 \\ (\nabla m_i)^{\perp} \\ 0 \\ \vdots \end{bmatrix}.$$

Also, for any $i < j = 1, \ldots, n$,

$$A_{\text{VTV}}(\mathbf{m}) \begin{bmatrix} \vdots \\ 0 \\ \nabla m_j \\ 0 \\ \vdots \\ 0 \\ -\nabla m_i \\ 0 \\ \vdots \end{bmatrix} = \frac{1}{|\nabla \mathbf{m}|} \begin{bmatrix} \vdots \\ 0 \\ \nabla m_j \\ 0 \\ \vdots \\ 0 \\ -\nabla m_i \\ 0 \\ \vdots \end{bmatrix}; \text{ indices: } \begin{Bmatrix} \vdots \\ i-1 \\ i \\ i+1 \\ \vdots \\ j-1 \\ j \\ j+1 \\ \vdots \end{Bmatrix}.$$

In the end, the eigenpairs for the diffusion tensor A_{VTV} are

$$\left(\begin{bmatrix}\nabla m_1\\ \vdots\\ \vdots\\ \nabla m_n\end{bmatrix}, 0\right), \left(\begin{bmatrix}\vdots\\ 0\\ (\nabla m_i)^{\perp}\\ 0\\ \vdots\end{bmatrix}, \frac{1}{|\nabla \mathbf{m}|}\right), \left(\begin{bmatrix}\vdots\\ 0\\ \nabla m_j\\ 0\\ \vdots\\ 0\\ -\nabla m_i\\ 0\\ \vdots\end{bmatrix}, \frac{1}{|\nabla \mathbf{m}|}\right).$$

The kernel of the diffusion tensor contains the search directions that will not be smoothed out, and will therefore be able to display sharp edges. Whereas there was no connections between all the different elements of the kernel in the case of the joint TV, the kernel of the diffusion tensor of VTV will favor parameter fields that vary sharply at the same physical locations.

4.2 Primal-dual Newton method for the solution of joint inverse problems regularized with VTV

As discussed in section 1.3.2, the use of a Newton-type method is prescribed for the solution of large-scale nonlinear inverse problems governed by PDEs, due to the high computational costs of the evaluation of the objective function, the computation of the gradient, or the computation of a Hessian-vector product. However, the use of the classical Newton method to solve a joint inverse problem regularized with VTV can require a large number of iterations and backtracking steps in the line search, and as a result a large number of PDE solves (see section 4.3); this can render the computational cost of large-scale applications prohibitively expensive. Building on the work of [19, 50], we propose instead a primal-dual Newton method.

In the rest of this chapter, we assume that for all $i = 1, \ldots, n$, $m_i \in H^1(\Omega)$, i.e., the Sobolev space of functions in $L^2(\Omega)$ with square integrable weak derivatives, and $\tilde{m}_i \in H_0^1(\Omega)$. Because Ω is a bounded domain, we have the embedding of the Lebesgue spaces $L^2(\Omega) \subset L^1(\Omega)$, and we can conclude that $H^1(\Omega) = W^{1,2}(\Omega) \subset W^{1,1}(\Omega)$, guaranteeing well-posedness of the VTV functional.

4.2.1 Formulation

To facilitate the presentation, we introduce a general joint inverse problem of type (1.35) or (1.36) regularized with VTV. The Lagrangian [82] for that prototypical joint inverse problem is defined as

$$\mathscr{L}(u, \mathbf{m}, p) = \mathcal{M}(u, d) + \gamma \int_\Omega \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}\, dx + \langle \mathcal{A}(u, \mathbf{m}), p \rangle, \tag{4.10}$$

where $\gamma > 0$, u (resp. p) is the state variable (resp. adjoint variable), $\mathcal{M}(u, d)$ denotes the data misfit part, and $\mathcal{A}(u, \mathbf{m})$ the PDE constraints. For instance, in the case of a joint inverse problem (1.35), u and p each represent a single variable, $\mathcal{M}(u, d) = \frac{1}{2}|Bu - \mathbf{d}|^2$, and $\langle \mathcal{A}(u, \mathbf{m}), p \rangle$ is the weak form of a single PDE constraint; in the case of a joint inverse problem (1.36), $u = \{u_i\}_i$, $p = \{p_i\}_i$, $\mathbf{d} = \{\mathbf{d}_i\}_i$, $\mathcal{M}(u, d) = \sum_{i=1}^n \frac{1}{2}|B_i u_i - \mathbf{d}_i|^2$, and $\langle \mathcal{A}(u, \mathbf{m}), p \rangle = \sum_{i=1}^n \langle \mathcal{A}_i(u_i, m_i), p_i \rangle$ is the sum of the n PDE constraints. The weak form of the gradient of the VTV joint regularization with respect to $\mathbf{m}$ is given by summing the individual contributions (4.7) to obtain

$$\delta_{\mathbf{m}} \mathcal{R}_{\text{VTV}}(\mathbf{m}; \tilde{\mathbf{m}}) = \sum_{k=1}^n \int_\Omega \frac{\nabla m_k \cdot \nabla \tilde{m}_k}{\sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}}\, dx.$$

The first-order optimality condition for the Lagrangian (4.10) is given by: Find $\mathbf{m}$ such that for all variations $\tilde{\mathbf{m}}$,

$$(\mathcal{G}(\mathbf{m}), \tilde{\mathbf{m}}) + \gamma \sum_{k=1}^{n} \int_{\Omega} \frac{\nabla m_k \cdot \nabla \tilde{m}_k}{\sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}}\, dx = 0, \tag{4.11}$$

where $\mathcal{G}$ is the derivative of the data-misfit part of the objective function, $\mathcal{M}(u(\mathbf{m}), d)$, with respect to $\mathbf{m}$.

We next introduce the dual variable $\mathbf{w} \coloneqq (w_1, \ldots, w_n)$, to approximate the quantities $\nabla m_k / \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}$, and its test function $\tilde{\mathbf{w}} \coloneqq (\tilde{w}_1, \ldots, \tilde{w}_n)$. We assume that for all $i = 1, \ldots, n$, $w_i, \tilde{w}_i \in [L^2(\Omega)]^d$. We re-write the first-order optimality condition (4.11) in primal-dual form as: Find $(\mathbf{m}, \mathbf{w})$ such that for all variations $(\tilde{\mathbf{m}}, \tilde{\mathbf{w}})$,

$$\begin{cases} (\mathcal{G}(\mathbf{m}), \tilde{\mathbf{m}}) + \gamma \displaystyle\sum_{k=1}^{n} \int_{\Omega} w_k \cdot \nabla \tilde{m}_k \, dx = 0, \\ \displaystyle\sum_{k=1}^{n} \int_{\Omega} (w_k \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon} - \nabla m_k) \cdot \tilde{w}_k \, dx = 0. \end{cases} \tag{4.12}$$

We then solve the first-order optimality condition (4.12) using Newton's method. The introduction of the dual variable $\mathbf{w}$ allows an equivalent expression of the first-order optimality system (4.11) that avoids the appearance of the terms $\nabla m_k / \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}$. As discussed in [19], Newton method applied to formulation (4.12) represents a more efficient linearization of the first-order optimality condition than the classical Newton method, and make the problem effectively less nonlinear. The Newton system for (4.12) is given by: Find $(\hat{\mathbf{m}}, \hat{\mathbf{w}})$ such

that for all variations $(\tilde{\mathbf{m}}, \tilde{\mathbf{w}})$,

$$\begin{cases} (\mathcal{H}_d(\mathbf{m})\hat{\mathbf{m}}, \tilde{\mathbf{m}}) + \\ \qquad \gamma \sum_{k=1}^{n} \int_\Omega \nabla \tilde{m}_k \cdot \hat{w}_k \, dx = -\left[(\mathcal{G}(\mathbf{m}), \tilde{\mathbf{m}}) + \gamma \sum_{k=1}^{n} \int_\Omega \nabla \tilde{m}_k \cdot w_k \, dx \right], \\ \sum_{k=1}^{n} \Big[\int_\Omega \tilde{w}_k \cdot \left(w_k \frac{\sum_{l=1}^{n} \nabla m_l \cdot \nabla \hat{m}_l}{\sqrt{|\nabla \mathbf{m}|^2 + \varepsilon}} - \nabla \hat{m}_k \right) dx + \\ \qquad \int_\Omega \tilde{w}_k \cdot \hat{w}_k \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon} \, dx \Big] = -\sum_{k=1}^{n} \int_\Omega \tilde{w}_k \cdot \left(w_k \sqrt{|\nabla \mathbf{m}|^2 + \varepsilon} - \nabla m_k \right) dx, \end{cases} \tag{4.13}$$

where $\mathcal{H}_d$ is the Hessian of the data-misfit part of the objective function, $\mathcal{M}(u(\mathbf{m}), d)$, with respect to $\mathbf{m}$.

4.2.2 Discretization

The joint inverse problem is discretized using the finite-element method. It is typical to use linear continuous Galerkin elements CG1 for the medium parameters $\{m_k\}_k$, and we follow that usage here. The dual variables $\{w_k\}_k$ are discretized with vector-valued constant discontinuous Galerkin elements $(\text{DG0})^d$. Let us call $\{\varphi_i\}_i$ a finite element basis for the space CG1. Let us call $\{\psi_i\}_i$ a basis for the vector-valued space $(\text{DG0})^d$. We denote with the subscript h the finite element approximations to the parameter fields introduced in the previous sections, i.e., $\mathbf{m}_h$ (resp. $m_{k,h}, w_{k,h}$) is the finite element approximation to the parameter field $\mathbf{m}$ (resp. m_k, w_k); we denote the vector containing all the finite-element coefficients of $m_{k,h}$ by $\mathbf{m}_k \coloneqq (m_{k,h}^1, m_{k,h}^2, \ldots)$, and we have the relation $m_{k,h} = \sum_i m_{k,h}^i \varphi_i$. Similar notations are used for the other parameter fields (e.g., $\hat{\mathbf{m}}$, $\hat{\mathbf{w}}$, $\mathbf{w}$). Let us call $\mathbf{H}_d$ the matrix for the data

misfit part of the Hessian $\mathcal{H}_d$. We also introduce the following notations,

$$\begin{aligned}
(\mathbf{H}_{\text{TV}})_{i,j} &\coloneqq \int_\Omega \nabla\varphi_i \cdot \psi_j \, dx, \\
(\mathbf{M}_w)_{i,j} &\coloneqq \int_\Omega \psi_i \cdot \psi_j \sqrt{|\nabla \mathbf{m}_h|^2 + \varepsilon} \, dx, \\
(\mathbf{B}(m_{k,h}, w_{k',h}))_{i,j} &\coloneqq \int_\Omega -\psi_i \cdot \left(w_{k',h} \frac{\nabla m_{k,h} \cdot \nabla \varphi_j}{\sqrt{|\nabla \mathbf{m}_h|^2 + \varepsilon}} \right) dx, \\
(\mathbf{g}_k)_i &\coloneqq \int_\Omega \psi_i \cdot \left(w_k \sqrt{|\nabla \mathbf{m}_h|^2 + \varepsilon} - \nabla m_k \right) dx.
\end{aligned} \tag{4.14}$$

For convenience, we also define the shorthand notation $\mathbf{B}_{k,k'} \coloneqq \mathbf{B}(m_{k,h}, w_{k',h})$.

Following discretization, the Newton system (4.13) becomes

$$\left[\begin{array}{ccc|cc}
 & \mathbf{H}_d(\mathbf{m}) & & \gamma\mathbf{H}_{\text{TV}} & 0 \\
 & & & \ddots & \\
 & & & 0 & \gamma\mathbf{H}_{\text{TV}} \\
\hline
-(\mathbf{H}_{\text{TV}}^T + \mathbf{B}_{1,1}) & & -\mathbf{B}_{n,1} & \mathbf{M}_w & 0 \\
 & \ddots & & \ddots & \\
-\mathbf{B}_{1,n} & & -(\mathbf{H}_{\text{TV}}^T + \mathbf{B}_{n,n}) & 0 & \mathbf{M}_w
\end{array}\right]
\left[\begin{array}{c}
\hat{\mathbf{m}}_1 \\ \vdots \\ \hat{\mathbf{m}}_n \\ \hline \hat{\mathbf{w}}_1 \\ \vdots \\ \hat{\mathbf{w}}_n
\end{array}\right]$$

$$= -\left[\begin{array}{c}
\mathbf{g}_{d,1} + \gamma\mathbf{H}_{\text{TV}}\mathbf{w}_1 \\ \vdots \\ \mathbf{g}_{d,n} + \gamma\mathbf{H}_{\text{TV}}\mathbf{w}_n \\ \hline \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_n
\end{array}\right]. \tag{4.15}$$

The Newton system in primal-dual form (4.15) comes from a saddle-point problem [19]. Although this could be handled directly, we instead reduce system (4.15) to its primal form by eliminating the dual variables $\hat{\mathbf{w}}_i$ from the equation, then employ a quasi-Newton method to recover a symmetric positive definite Hessian for the VTV term. Because $w_{k,h}$ is piecewise constant, the matrix $\mathbf{M}_w$ is diagonal. Using that fact, we can easily compute the expression

of the variables $(\hat{\mathbf{w}}_1, \dots, \hat{\mathbf{w}}_n)$ in (4.15),

$$\begin{bmatrix} \hat{\mathbf{w}}_1 \\ \vdots \\ \hat{\mathbf{w}}_n \end{bmatrix} = - \begin{bmatrix} \mathbf{M}_w^{-1}\mathbf{g}_1 \\ \vdots \\ \mathbf{M}_w^{-1}\mathbf{g}_n \end{bmatrix} + \begin{bmatrix} \mathbf{M}_w^{-1} & & 0 \\ & \ddots & \\ 0 & & \mathbf{M}_w^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{\text{TV}}^T + \mathbf{B}_{1,1} & & \mathbf{B}_{n,1} \\ & \ddots & \\ \mathbf{B}_{1,n} & & \mathbf{H}_{\text{TV}}^T + \mathbf{B}_{n,n} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{m}}_1 \\ \vdots \\ \hat{\mathbf{m}}_n \end{bmatrix}.$$

We use this expression to eliminate the variables $(\hat{\mathbf{w}}_1, \dots, \hat{\mathbf{w}}_n)$ in (4.15), and we obtain the reduced Newton's system in terms of the variables $(\hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2, \dots)$,

$$(\mathbf{H}_d(\mathbf{m}) + \mathbf{H}_{\text{VTV}}) \begin{bmatrix} \hat{\mathbf{m}}_1 \\ \vdots \\ \hat{\mathbf{m}}_n \end{bmatrix} = - \begin{bmatrix} \mathbf{g}_{d,1} + \gamma \mathbf{H}_{\text{TV}}(\mathbf{w}_1 - \mathbf{M}_w^{-1}\mathbf{g}_1) \\ \vdots \\ \mathbf{g}_{d,n} + \gamma \mathbf{H}_{\text{TV}}(\mathbf{w}_n - \mathbf{M}_w^{-1}\mathbf{g}_n) \end{bmatrix}, \tag{4.16}$$

where we introduced the notation $\mathbf{H}_{\text{VTV}}$, for the reduced Hessian of the VTV functional, defined as

$$\mathbf{H}_{\text{VTV}} \coloneqq \begin{bmatrix} \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} & & 0 \\ & \ddots & \\ 0 & & \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{\text{TV}}^T + \mathbf{B}_{1,1} & & \mathbf{B}_{n,1} \\ & \ddots & \\ \mathbf{B}_{1,n} & & \mathbf{H}_{\text{TV}}^T + \mathbf{B}_{n,n} \end{bmatrix}. \tag{4.17}$$

In (4.16), the reduced Hessian $\mathbf{H}_{\text{VTV}}$ is neither guaranteed to be symmetric, nor positive definite. Although this could be handled by appropriate methods, it is more efficient to try and preserve the structure of the VTV functional, which naturally leads to a symmetric positive definite Hessian. We decompose the reduced Hessian $\mathbf{H}_{\text{VTV}}$ (4.17) into a symmetric positive definite term and the rest which we denote by $\mathbf{H}_w$,

$$\mathbf{H}_{\text{VTV}} = \begin{bmatrix} \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \mathbf{H}_{\text{TV}}^T & & 0 \\ & \ddots & \\ 0 & & \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \mathbf{H}_{\text{TV}}^T \end{bmatrix} + \mathbf{H}_w(\mathbf{w}_h),$$

with

$$\mathbf{H}_w(\mathbf{w}_h) \coloneqq \begin{bmatrix} \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} & & 0 \\ & \ddots & \\ 0 & & \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{1,1} & & \mathbf{B}_{n,1} \\ & \ddots & \\ \mathbf{B}_{1,n} & & \mathbf{B}_{n,n} \end{bmatrix}.$$

We obtain the Newton system of the primal-dual Newton method, by replacing $\mathbf{H}_{\text{VTV}}$ in (4.16) with a symmetric positive definite approximation $\bar{\mathbf{H}}_{\text{VTV}}$,

$$\left(\mathbf{H}_d(\mathbf{m}) + \bar{\mathbf{H}}_{\text{VTV}}\right) \begin{bmatrix} \hat{\mathbf{m}}_1 \\ \vdots \\ \hat{\mathbf{m}}_n \end{bmatrix} = - \begin{bmatrix} \mathbf{g}_{d,1} + \gamma \mathbf{H}_{\text{TV}}(\mathbf{w}_1 - \mathbf{M}_w^{-1}\mathbf{g}_1) \\ \vdots \\ \mathbf{g}_{d,n} + \gamma \mathbf{H}_{\text{TV}}(\mathbf{w}_n - \mathbf{M}_w^{-1}\mathbf{g}_n) \end{bmatrix}. \tag{4.18}$$

The matrix $\bar{\mathbf{H}}_{\text{VTV}}$ is defined as

$$\bar{\mathbf{H}}_{\text{VTV}} = \begin{bmatrix} \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \mathbf{H}_{\text{TV}}^T & & 0 \\ & \ddots & \\ 0 & & \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \mathbf{H}_{\text{TV}}^T \end{bmatrix} + \frac{1}{2}\left(\mathbf{H}_w(\bar{\mathbf{w}}_h) + \mathbf{H}_w^T(\bar{\mathbf{w}}_h)\right),$$

where $\bar{\mathbf{w}}_h \coloneqq (\bar{w}_{1,h}, \ldots, \bar{w}_{n,h})$ and

$$\mathbf{H}_w(\bar{\mathbf{w}}_h) \coloneqq \begin{bmatrix} \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} & & 0 \\ & \ddots & \\ 0 & & \gamma \mathbf{H}_{\text{TV}} \mathbf{M}_w^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{B}(m_{1,h}, \bar{w}_{1,h}) & & \mathbf{B}(m_{n,h}, \bar{w}_{1,h}) \\ & \ddots & \\ \mathbf{B}(m_{1,h}, \bar{w}_{n,h}) & & \mathbf{B}(m_{n,h}, \bar{w}_{n,h}) \end{bmatrix}.$$

The re-scaled dual variables, $\bar{w}_{i,h} = \sum_j \bar{w}_i^j \psi_j$, are defined through their finite element coefficients, $\bar{w}_i^j = (\bar{w}_{i,1}^j, \ldots, \bar{w}_{i,d}^j) \in \mathbb{R}^d$, by the formula

$$\bar{w}_i^j \coloneqq \frac{w_i^j}{\max(1, \sqrt{\sum_{k=1}^n |w_k^j|^2})}, \quad \forall i = 1, \ldots, n,$$

where $|w_k^j|^2 = \sum_{s=1}^d (w_{k,s}^j)^2$ is the l^2-norm of the j^{th} finite-element coefficient of w_k. The re-scaled dual variables verify $\sum_{i=1}^n |\bar{w}_i^j|^2 \leq 1$. This choice of the scaling factor can be justified with a duality argument (see, for instance, the P2 elements in [14]). We also verified numerically that the choice of the unitary l^2-ball guaranteed positive definiteness, and was optimal.

4.2.3 Numerical solution

We solve the Newton system (4.18) using the preconditioned conjugate gradient method [67]. While the regularization Hessian $\bar{\mathbf{H}}_{\text{VTV}}$ is guaranteed positive definite, the full Hessian (including the data misfit term $\mathbf{H}_d$) can lead to directions of negative curvature away from a minimum. We use early truncation of the Krylov iterations to cope with this situation and to guarantee a descent direction at each nonlinear iteration [27]. To limit the cost of the line search, we use a backtracking line search [67].

Since we solve the linear systems (4.18) that arise at each Newton iteration using the conjugate gradient method, it is essential to have both a scalable and effective preconditioner. The Hessian of the regularization $\bar{\mathbf{H}}_{\text{VTV}}$ usually has full rank, while the data misfit Hessian $\mathbf{H}_d$ is often compact with an (approximately) finite-dimensional range space. Thus preconditioning by approximately inverting the Hessian of the regularization term produces a "compact + identity" structure for which a Krylov method converges in a mesh-independent number of iterations. In practice, we found a BFGS approximation to the full Hessian, initialized by the inverse of the regularization Hessian $\bar{\mathbf{H}}_{\text{VTV}}$ performs better than the use of the regularization Hessian alone. Since the full Hessian is not guaranteed to be positive definite, we use a damped update to build our BFGS preconditioner; more details can be found in section 3.3.4. Remains the question of applying the inverse of the Hessian of the VTV functional. By construction, in the primal-dual Newton algorithm, $\bar{\mathbf{H}}_{\text{VTV}}$ is symmetric and positive definite, and has the structure of a (vector) anisotropic elliptic PDE operator. Thus, a few V-cycles of an algebraic multigrid method is sufficient to approximately invert the regularization Hessian, and we have used this approach for the results reported here.

4.3 Numerical examples

We verify scalability of the primal-dual Newton method with respect to the regularization hyperparameter ε, the mesh size h (defined in section 3.4), and the number of inversion parameters n. The examples use the joint acoustic inverse problem (1.28) and a joint Poisson inverse problem combining n independent Poisson inverse problems (3.20), and cover both types of joint inverse problems (1.35) and (1.36). We now present the joint inverse problems used in this section.

The joint acoustic wave problem (1.28), introduced in section 1.4.2, is defined in terms of the inverse of the bulk modulus λ and the density ρ which we denote by $\alpha \coloneqq 1/\lambda$ and $\beta \coloneqq 1/\rho$. We use 3 independent forcing terms of frequency 1.5Hz, located at (0.1, 0.1), (0.5, 0.1), and (0.9, 0.1). The acoustic wave inverse problem regularized with VTV is an example of joint inverse problem (1.35). The formulation of the independent Poisson inverse problem regularized with TV for parameter m_i, for any $i = 1, \ldots, n$, is given by (3.20). The joint Poisson inverse problem is a joint inverse problem (1.36) for n inverse problems (3.20). The n inverse problems are uncoupled, but to justify the use of a joint inverse problem, we assume that their truth parameter fields are known to share structure.

In all examples, the observation operators B and B_i are pointwise observation operators, and the data $\mathbf{d}_i$ are synthetic observations computed at the truth parameter field for m_i, then polluted with additive, independent and identically distributed Gaussian noise. The noise level for the Poisson inverse problem (3.20) is set to 2%. The variance of the noise for the acoustic wave inverse problem (1.28) is set to have a noise-to-signal ratio of 20 dB. The domain is a unit square, meshed with a structured grid of identical isosceles

square triangles. The minimization problem and the PDE constraints are discretized with the finite-element method using continuous Galerkin elements; for the state, adjoint, incremental state, and incremental adjoint variables we use quadratic elements, and for the parameter fields we use linear elements. The values of the regularization parameter γ for VTV in all joint inverse problems are selected empirically as leading to the best reconstructions. For all examples, the numerical solution is implemented in Python with the finite-element library FEniCS [61, 62]. For the Poisson joint inverse problems, the examples are solved using the optimization routines from hIPPYlib [84], a Python library for deterministic and Bayesian inverse problems.

4.3.1 Scalability with respect to the hyperparameter ε

We show how the performance of the primal-dual Newton solver varies with the hyperparameter ε in two examples. This hyperparameter introduces a trade-off between sharp edges in the reconstruction and a smooth regularization; the smaller the hyperparameter ε, the more nonlinear the inverse problem becomes, and thus, the more forward PDE solves it may require. We compare the performance and scalability of the primal-dual Newton solver with the primal Newton solver (see section 1.3.2) and the lagged diffusivity. The lagged diffusivity consists of freezing the values of the parameter fields m_i at their current estimate, inside the Hessian of the VTV regularization [86]. It is a Picard method, and as such only has a linear rate of convergence, but it is popular in applications [19, 87].

4.3.1.1 Joint Poisson inverse problem

We solve a joint inverse problem (3.20) with two parameters m_1 and m_2. The mesh, the truth parameter fields (section 4.1(0)), and the observation operators are as in the joint Poisson inverse problem of section 3.4.1.1. The solution of two independent inverse problems (3.20) leads to a very good reconstruction for m_2 and a poor one for m_1. In particular, the edges in the truth parameter field for m_1 (figure 4.1a(0)) are only visible in the top right quadrant of the reconstruction (figure 3.5a).

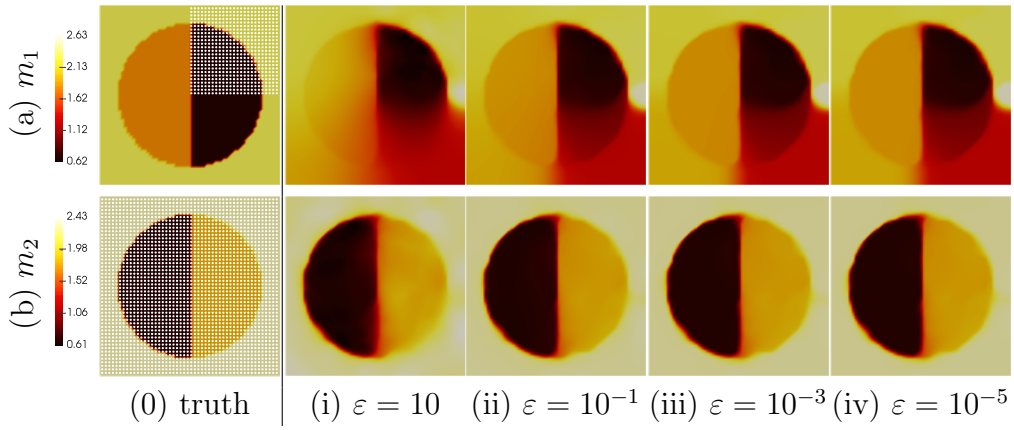


Figure 4.1: Plots of the (0) truth and pointwise observations (white dots), and (i-iv) reconstructions for parameter fields (a) m_1 and (b) m_2, obtained by solving the joint Poisson inverse problem (3.20) regularized with VTV ($\gamma = 3 \cdot 10^{-7}$) with hyperparameter (i) $\varepsilon = 10$, (ii) $\varepsilon = 10^{-1}$, (iii) $\varepsilon = 10^{-3}$, and (iv) $\varepsilon = 10^{-5}$.

In order to transfer some structural information from the reconstruction of m_2 onto m_1, we solve a joint Poisson inverse problem (1.36) for m_1 and m_2 regularized with VTV. We show the reconstructions obtained for different values of the hyperparameter ε in figure 4.1(i-iii), and the relative medium misfits are listed in table C.1. As the hyperparameter ε is set smaller, the

reconstructions become sharper (compare figures 4.1b(i) and 4.1b(iv)), and the reconstructions for m_1 capture more and more of the originally missing edges (compare figures 4.1a(i) and 4.1a(iv)).

In figure 4.2, we plot the norm of the gradient of the joint Poisson inverse problem against the number of forward PDE solves performed, for different values of the hyperparameter ε, and using three different solvers, namely the primal Newton method, the lagged diffusivity method, and the primal-dual Newton method introduced in section 4.2. As the hyperparameter ε is set

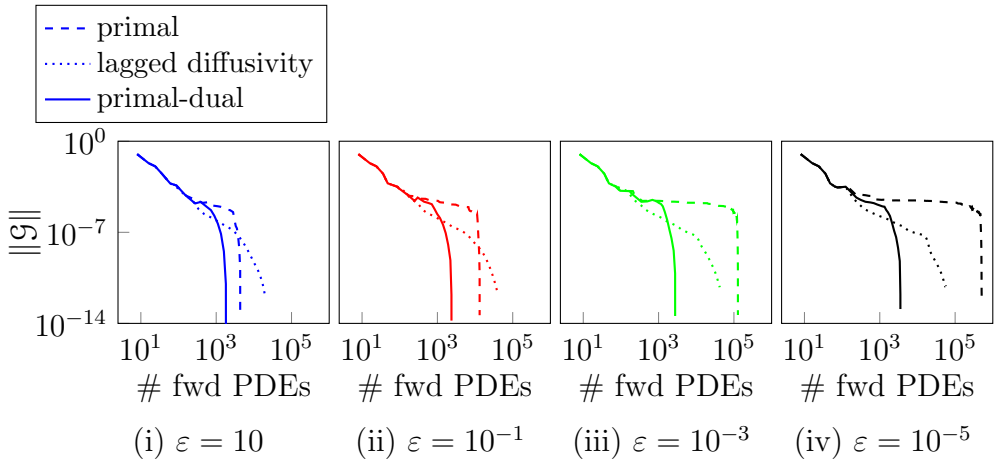


Figure 4.2: Plots of the norm of the gradient against the number of forward PDE solves performed during the solution of the joint Poisson inverse problem (3.20) regularized with VTV ($\gamma = 3 \cdot 10^{-7}$), and solved using the Newton method (dashed), the lagged diffusivity method (dotted), and the primal-dual Newton method (solid). The hyperparameter in VTV is set to (i) $\varepsilon = 10$, (ii) $\varepsilon = 10^{-1}$, (iii) $\varepsilon = 10^{-3}$, and (iv) $\varepsilon = 10^{-5}$.

smaller, the number of PDE solves required to solve the joint inverse problem with the primal Newton method and the lagged diffusivity method grows by orders of magnitudes. In figure 4.3, we show the norm of the gradient against the number of forward PDE solves for the different values of ε but only for

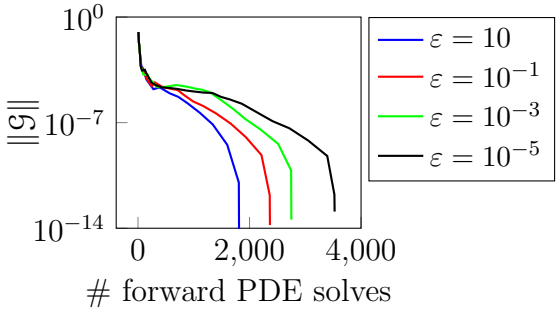


Figure 4.3: Plot of the norm of the gradient against the number of forward PDE solves performed in the solution of the joint inverse problem (3.20) regularized with VTV, and solved using the primal-dual Newton method.

the primal-dual Newton method. The number of PDE solves required in the primal-dual Newton method grows mildly as the hyperparameter ε becomes smaller.

4.3.1.2 Joint inversion of bulk modulus and density in the acoustic wave equation

We study scalability of the primal-dual Newton method with respect to the hyperparameter ε in the joint inverse problem (1.28). The mesh contains 3200 isosceles square triangles ($h = 1/40$). The initial value of the parameter fields are chosen as the background values of the truth parameter fields (see figure 4.4(0)). The locations of the sources and receivers are shown in figure 4.4(0)), along with the reconstructions for parameters α and β, corresponding to increasingly smaller hyperparameters ε (see figure 4.4(i-iii)). The relative medium misfits corresponding to these reconstructions are given in table C.2.

In figure 4.5, we plot the norm of the gradient of the joint inverse problem (1.28) again the number of forward PDE solves performed. We observed

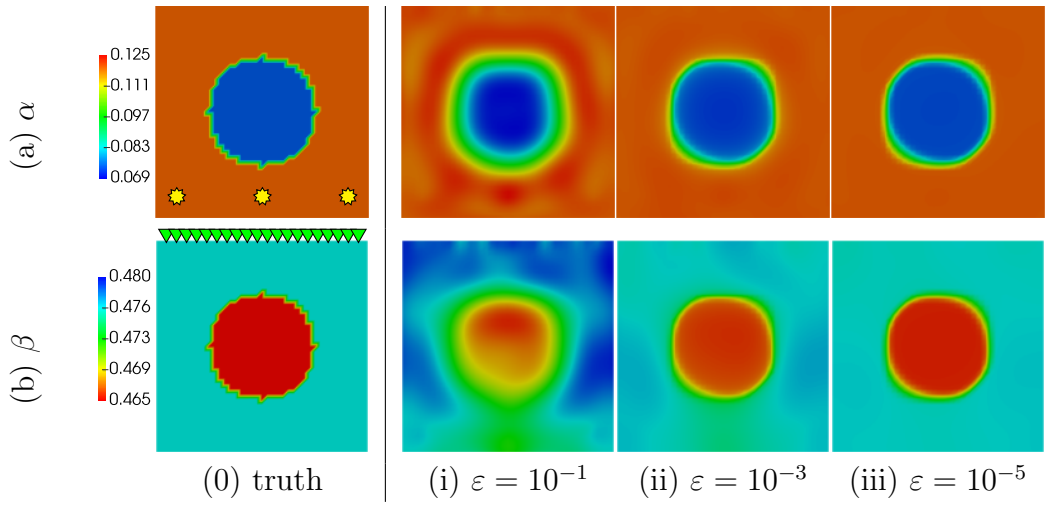


Figure 4.4: Plots of the (0) truth and (i-iii) reconstructions for parameter fields (a) α and (b) β, obtained by solving the acoustic wave inverse problem (1.28) regularized with VTV ($\gamma = 4 \cdot 10^{-7}$). The hyperparameter ε in VTV is set to (i) $\varepsilon = 10^{-1}$, (ii) $\varepsilon = 10^{-3}$, and (iii) $\varepsilon = 10^{-5}$. The locations of the sources (yellow stars) and pointwise observations (green triangles) are plotted in (0).

similar results as for the joint Poisson inverse problem (see section 4.3.1.1), i.e., the number of PDE solves required by the primal Newton method and the lagged diffusivity method grow at a much larger rate than with the primal-dual Newton method. In figure 4.5(iv), we plot the number of PDE solves for the primal-dual Newton method only. Although this number does grow as the hyperparameter ε is made smaller, the dependence is mild.

4.3.2 Scalability with respect to the mesh size h

We use the joint Poisson inverse problem (3.20) to illustrate scalability of the primal-dual Newton method with respect to the mesh size. The regularization hyperparameter is set to $\varepsilon \equiv 10^{-3}$, and we vary the mesh parameter h. The results are shown in figure 4.6.

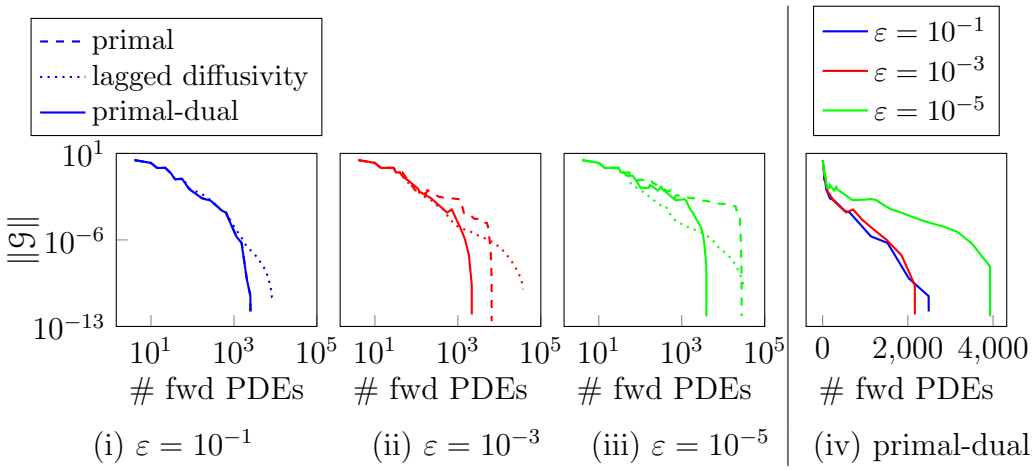


Figure 4.5: Plots of the norm of the gradient against the number of forward PDE solves performed during the solution of the acoustic wave inverse problem (1.28) regularized with VTV ($\gamma = 4 \cdot 10^{-7}$), and solved using the Newton method (dashed), the lagged diffusivity method (dotted), and the primal-dual Newton method (solid). The hyperparameter ε in VTV is set to (i) $\varepsilon = 10^{-1}$, (ii) $\varepsilon = 10^{-3}$, (iii) $\varepsilon = 10^{-5}$. The numbers of forward PDE solves for the primal-dual Newton only, with all values of ε, are plotted in (iv).

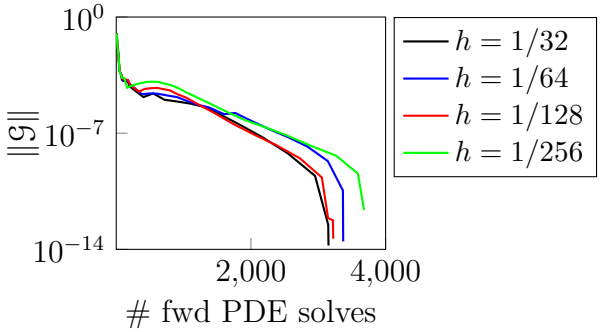


Figure 4.6: Plot of the norm of the gradient against the number of forward PDE solves performed during the solution of the joint Poisson inverse problem (3.20) regularized with VTV ($\gamma = 3 \cdot 10^{-7}$ and $\varepsilon = 10^{-3}$), and solved with the primal-dual Newton method. The realization of the noise is different from section 4.3.1. The mesh parameter h was defined in section 3.4.

The number of forward PDE solves required to solve the joint inverse problem (3.20) regularized with VTV grows mildly as the mesh gets refined. Such mild growth should however be expected, here. The magnitude of the norm of the discrete gradients inside the VTV functional grows in the order of $1/h^2$. Therefore since the hyperparameter ε is kept constant while the mesh is being refined, the inverse problem becomes more nonlinear. As the exact relationship between mesh refinement and hyperparameter ε is not known, we instead present the results for a constant hyperparameter ε.

4.3.3 Scalability with respect to the number of inversion parameters n

We show scalability of the primal-dual Newton method with respect to the number of inversion parameters n on the Poisson joint inverse problem (3.20). The mesh contains 8192 isosceles square triangles ($h = 1/64$). The observations for parameter m_i are pointwise observations at 160 locations

randomly sampled from a square lattice of 50×50 pointwise observations covering the entire domain, with higher concentrations in one of the quadrants (100 observations in this quadrant vs 20 elsewhere); observations for m_i, for $i = 1, 5, 9, 13$ (resp. $i = 2, 6, 10, 14$; $i = 3, 7, 11, 15$; $i = 4, 8, 12, 16$), have a higher number of observations in the top-left (resp. bottom-left; bottom-right; top-right) quadrant. We show the truth parameter fields for each parameter field m_i on figure 4.7.

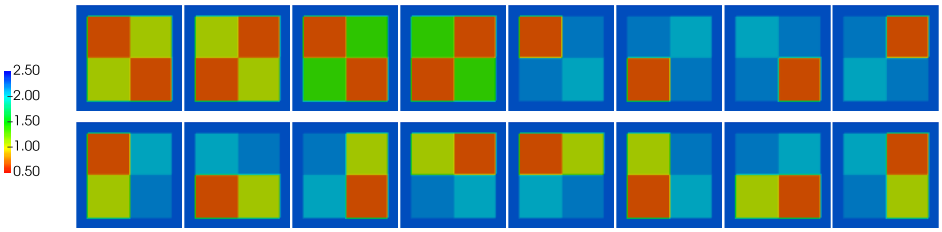


Figure 4.7: Truth parameter fields for parameters m_i, $i = 1, \ldots, 16$ (reading from left to right, from top to bottom), for the example in section 4.3.3.

Next we solve a sequence of joint Poisson inverse problems (3.20) regularized with VTV for n parameter fields $(m_1, \ldots, m_n)$ using the primal-dual Newton method, with n varying from 1 to 16. As the number of inversion parameters n gets bigger, the effective value of the hyperparameter ε on the solution of the joint inverse problem, i.e., its value against the norm of the n gradients in the VTV functional, will become smaller. Therefore keeping ε constant as the number of inversion parameters gets bigger will make the problem more nonlinear. In order to maintain the influence of the hyperparameter ε constant across all joint inverse problems, we normalize the contribution of the gradients in the expression for VTV (4.6) to obtain $\mathcal{R}_{\text{VTV}}(m) = \int_\Omega \sqrt{\frac{1}{n}\sum_{i=1}^n |\nabla m_i|^2 + \varepsilon}\, dx$, or equivalently $\mathcal{R}_{\text{VTV}}(m) = \frac{1}{\sqrt{n}} \int_\Omega \sqrt{\sum_{i=1}^n |\nabla m_i|^2 + n\varepsilon}\, dx$. In the latter formulation, the factor $1/\sqrt{n}$ can

be absorbed by the regularization parameter γ, leaving the expression

$$\mathcal{R}_{\text{VTV}}(m) = \int_\Omega \sqrt{\sum_{i=1}^{n} |\nabla m_i|^2 + \varepsilon_n}\, dx,$$

where we introduced the quantity $\varepsilon_n \coloneqq n\varepsilon_1$. In this example we used $\varepsilon_1 = 10^{-3}$. In figure 4.8, we show the reconstructions for parameter field m_1 obtained by solving the joint inverse problem (3.20) with n parameter fields. In table C.3, we show the relative medium misfits for each joint inverse problem. The quality of the reconstructions, for all parameter fields, increases with the

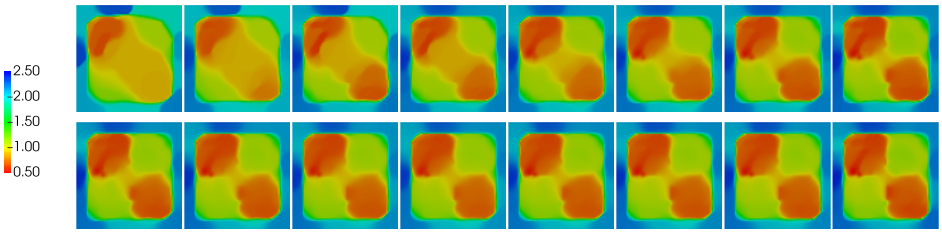


Figure 4.8: Reconstructions for parameter field m_1 obtained from solving a joint inverse problem with n parameters, $n = 1, \ldots, 16$ (reading from left to right, from top to bottom), for the example in section 4.3.3.

number of inversion parameters n, indicating that more information is added to the joint inverse problem.

In figure 4.9a, we compare how quickly the joint inverse problem converges when the number of inversion parameters n increases. In figure 4.9b, we plot the final number of PDE solves per number of inversion parameters n, at the minimizer, against the number of inversion parameters n, and compare this plot with a linear growth of that quantity. The number of forward PDE solves, despite our rescaling of the hyperparameter ε, grows at a constant rate; however this growth is moderate. On the other hand, as the number of

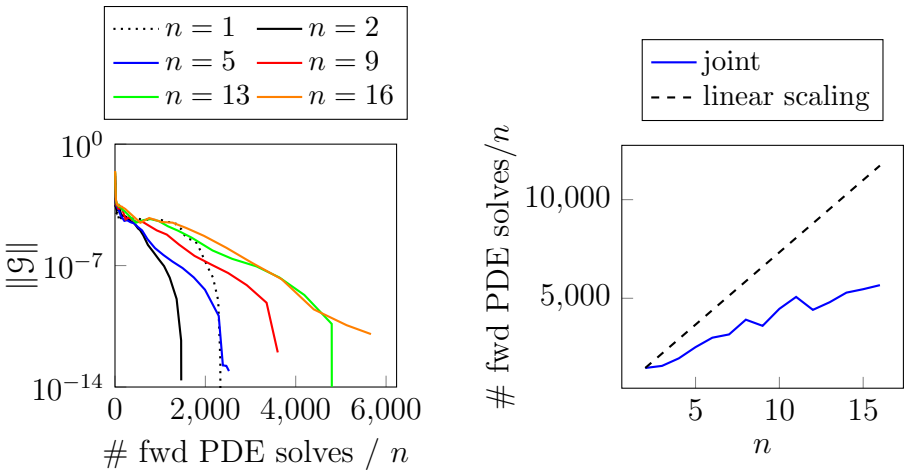


Figure 4.9: Scalability of the primal-dual Newton method with respect to the number of inversion parameters n. (a) Plot of the norm of the gradient against the number of forward PDE solves per number of inversion parameters n, for the solution of the joint Poisson inverse problem (3.20) regularized with VTV ($\varepsilon = 10^{-3} \times n$) and solved with the primal-dual Newton method. (b) Plot of the number of forward PDE solves per number of inversion parameters n, at the minimizer, against the number of inversion parameters n.

inversion parameters n grows, the number of Newton steps it takes to solve the joint inverse problem remains nearly constant (see figure 4.10). Hence, this growth in the number of forward PDE solves can be explained by the additional information being added to the joint inverse problem. As more information is added to the joint inverse problem, the rank of the Hessian increases, which translates into a higher number of conjuage gradient iterations required to solve the Newton system, and therefore a higher number of forward PDE solves.

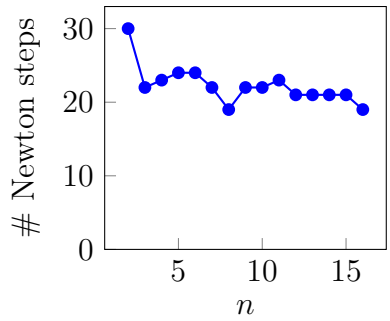


Figure 4.10: Plot of the number of Newton steps required to solve the joint Poisson inverse problem (3.20) regularized with VTV against the number of inversion parameters n.

4.4 Conclusion

The vectorial total variation joint regularization is an extension of total variation to the case of multi-parameter joint inverse problems. It originated from the imaging community, but was found to perform well for joint inverse problems constrained by PDEs (see section 3). In this chapter, we developed a primal-dual Newton method for the solution of joint inverse problems regularized with the vectorial total variation. This primal-dual Newton method introduces an auxiliary variable to construct a more efficient linearization of the first-order optimality condition. To make the method amenable for large-scale problems, we use a quasi-Newton method, replacing the reduced Hessian of the primal-dual formulation with a symmetric positive definite approximation. As the method converges, this approximation becomes negligible. We demonstrated numerically through a comprehensive set of examples that the primal-dual Newton solver scales with respect to the mesh size, the hyperparameter ε, and the number of inversion parameters. In addition, for practical situations, the primal-dual Newton method significantly outperforms the primal Newton method and the lagged diffusivity approach.

Appendices

Appendix A

Appendix for chapter 2

A.1 Gradient of the optimization formulation (2.20)

We detail the derivation of the gradient of the Laplace formulation of the A-optimal weights in the case of the Helmholtz inverse problem, as defined in (2.20). In that formulation, we enforce the PDE constraints weakly using Lagrange multipliers. Therefore, we need to introduce adjoint variables that are indicated with a star superscript, e.g., m^* is the adjoint variable for m. Following the formal Lagrangian approach [82], we define the Lagrangian $\mathscr{L}$,

$$
\begin{aligned}
&\mathscr{L}(\mathbf{w}, m, \{u_i\}, \{p_i\}, \{v_{i,k}\}, \{q_{i,k}\}, \{y_k\}, m^*, \{u_i^*\}, \{p_i^*\}, \{v_{i,k}^*\}, \{q_{i,k}^*\}, \{y_k^*\}) = \\
&\frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \langle y_k, z_k \rangle + \\
&\frac{1}{n_{tr} N_w} \sum_{k=1}^{n_{tr}} \sum_{i=1}^{N_w} \left[\langle \nabla v_{i,k}, \nabla v_{i,k}^* \rangle - \kappa^2 \langle m v_{i,k}, v_{i,k}^* \rangle - \kappa^2 \langle u_i y_k, v_{i,k}^* \rangle \right] \\
&+ \frac{1}{n_{tr} N_w} \sum_{k=1}^{n_{tr}} \sum_{i=1}^{N_w} \Bigg[\langle \nabla q_{i,k}^*, \nabla q_{i,k} \rangle - \kappa^2 \langle q_{i,k}^*, m q_{i,k} \rangle - \kappa^2 \langle q_{i,k}^*, p_i y_k \rangle \\
&\qquad\qquad + \langle B q_{i,k}^*, B v_{i,k} \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} \Bigg] \\
&+ \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \left[\langle y_k, y_k^* \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \left(\langle v_{i,k} p_i, y_k^* \rangle + \langle u_i q_{i,k}, y_k^* \rangle \right) - \langle z_k, y_k^* \rangle \right] +
\end{aligned}
$$

$$\frac{1}{N_w}\sum_{i=1}^{N_w}\left[\langle\nabla u_i,\nabla u_i^*\rangle-\kappa^2\langle mu_i,u_i^*\rangle-\langle f(\mathbf{w}^i),u_i^*\rangle\right]$$
$$+\frac{1}{N_w}\sum_{i=1}^{N_w}\left[\langle\nabla p_i^*,\nabla p_i\rangle-\kappa^2\langle p_i^*,mp_i\rangle+\langle Bp_i^*,Bu_i-\mathbf{d}(\mathbf{w}^i)\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}}\right]$$
$$+\langle m-m_0,m^*\rangle_{\mathbb{E}}-\frac{1}{N_w}\sum_{i=1}^{N_w}\kappa^2\langle u_ip_i,m^*\rangle. \tag{A.1}$$

The gradient is then given by $\delta_{\mathbf{w}}\mathscr{L}=[\delta_{\mathbf{w}^1}\mathscr{L},\delta_{\mathbf{w}^2}\mathscr{L},\ldots,\delta_{\mathbf{w}^{Nw}}\mathscr{L}]^T$, where for any $i=1,\ldots,N_w$,

$$\delta_{\mathbf{w}^i}\mathscr{L}=-\frac{1}{N_w}\begin{bmatrix}\langle f_1,u_i^*\rangle & + & \langle Bp_i^*,\mathbf{d}_1\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}}\\ \langle f_2,u_i^*\rangle & + & \langle Bp_i^*,\mathbf{d}_2\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}}\\ & \vdots & \\ \langle f_{N_s},u_i^*\rangle & + & \langle Bp_i^*,\mathbf{d}_{N_s}\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}}\end{bmatrix}.$$

Before we specify the steps that lead to the evaluation of the variables u_i^* and p_i^*, we identify some important symmetries between the state variables and their adjoints. Indeed, for each $k=1,\ldots,n_{tr}$, the variables $(y_k\{v_{i,k}\}_i,\{q_{i,k}\}_i)$ solve a Hessian system similar to (2.19), and the corresponding adjoint variables $(y_k^*\{v_{i,k}^*\}_i,\{q_{i,k}^*\}_i)$ solve the system of equations given (formally) by $\delta_{v_{ik}}\mathscr{L}=\delta_{q_{ik}}\mathscr{L}=\delta_{y_k}\mathscr{L}=0$. While the former system of equations solve $\mathcal{H}y_k=z_k$, the latter solves $\mathcal{H}y_k^*=-z_k$. This leads to the symmetry relations

$$y_k=-y_k^*,\ v_{ik}=-q_{ik}^*,\ \text{and}\ q_{ik}=-v_{ik}^*, \tag{A.2}$$

for any $i=1,\ldots,N_w$ and $k=1,\ldots,n_{tr}$.

For any $i=1,\ldots,N_w$, the variable u_i^* (resp. p_i^*) solves the equation $\delta_{u_i}\mathscr{L}=0$ (resp. $\delta_{p_i}\mathscr{L}=0$). That is, for any $\tilde{u}\in H^1(\Omega)$, u_i^* solves

$$\langle\nabla u_i^*,\nabla\tilde{u}\rangle-\kappa^2\langle mu_i^*,\tilde{u}\rangle$$
$$-\kappa^2\langle p_im^*,\tilde{u}\rangle+\langle Bp_i^*,B\tilde{u}\rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}}-\kappa^2\frac{1}{n_{tr}}\sum_{k=1}^{n_{tr}}\left[\langle y_kv_{i,k}^*,\tilde{u}\rangle+\langle q_{i,k}y_k^*,\tilde{u}\rangle\right]=0.$$

On the other hand, for any $\tilde{p} \in H^1(\Omega)$, p_i^* solves

$$\langle \nabla p_i^*, \nabla \tilde{p} \rangle - \kappa^2 \langle p_i^*, m\tilde{p} \rangle - \kappa^2 \langle u_i m^*, \tilde{p} \rangle - \kappa^2 \frac{1}{n_{tr}} \sum_{k=1}^{n_{tr}} \left[\langle q_{i,k}^* y_k, \tilde{p} \rangle + \langle v_{i,k} y_k^*, \tilde{p} \rangle \right] = 0.$$

Using (A.2), this reduces, for any $i = 1, \ldots, N_w$, to the system of equations

$$\begin{aligned} \langle \nabla u_i^*, \nabla \tilde{u} \rangle - \kappa^2 \langle m u_i^*, \tilde{u} \rangle - \kappa^2 \langle p_i m^*, \tilde{u} \rangle + \\ \langle B p_i^*, B\tilde{u} \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} + \frac{2}{n_{tr}} \sum_{k=1}^{n_{tr}} \kappa^2 \langle y_k q_{i,k}, \tilde{u} \rangle = 0, \\ \langle \nabla p_i^*, \nabla \tilde{p} \rangle - \kappa^2 \langle m p_i^*, \tilde{p} \rangle - \kappa^2 \langle u_i m^*, \tilde{p} \rangle + \frac{2}{n_{tr}} \sum_{k=1}^{n_{tr}} \kappa^2 \langle v_{i,k} y_k, \tilde{p} \rangle = 0. \end{aligned} \tag{A.3}$$

Therefore, computation of the u_i^*'s and p_i^*'s requires knowledge of the quantities $\{u_i\}$, $\{p_i\}$, m^*, $\{v_{i,k}\}$, $\{q_{i,k}\}$ and $\{y_k\}$. Variables $\{u_i\}$, $\{p_i\}$, $\{v_{i,k}\}$, $\{q_{i,k}\}$, and $\{y_k\}$ are all evaluated during the computation of the objective functional $1/n_{tr} \sum_{k=1}^{n_{tr}} \langle y_k, z_k \rangle$, such that the only remaining unknown quantity is m^*. That variable is solution to the equation $\delta_m \mathscr{L} = 0$, that is, for any $\tilde{m} \in \mathbb{E}$, m^* solves

$$\begin{aligned} \frac{1}{n_{tr} N_w} \sum_{k=1}^{n_{tr}} \sum_{i=1}^{N_w} \left[-\kappa^2 \langle \tilde{m} v_{i,k}, v_{i,k}^* \rangle - \kappa^2 \langle q_{i,k}^*, \tilde{m} q_{i,k} \rangle \right] \\ + \frac{1}{N_w} \sum_{i=1}^{N_w} \left[-\kappa^2 \langle \tilde{m} u_i, u_i^* \rangle - \kappa^2 \langle p_i^*, \tilde{m} p_i \rangle \right] + \langle \tilde{m}, m^* \rangle_{\mathbb{E}} = 0. \end{aligned}$$

Using (A.2), we simplify this equation to obtain

$$\frac{2}{n_{tr} N_w} \sum_{k=1}^{n_{tr}} \sum_{i=1}^{N_w} \kappa^2 \langle v_{i,k} q_{i,k}, \tilde{m} \rangle - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \left[\langle u_i u_i^*, \tilde{m} \rangle + \langle p_i^* p_i, \tilde{m} \rangle \right] + \langle m^*, \tilde{m} \rangle_{\mathbb{E}} = 0.$$

This equation can be grouped with the system of equations (A.3) to obtain

the larger system

$$\langle \nabla p_i^*, \nabla \tilde{p} \rangle - \kappa^2 \langle m p_i^*, \tilde{p} \rangle - \kappa^2 \langle u_i m^*, \tilde{p} \rangle = -\frac{2}{n_{tr}} \sum_{k=1}^{n_{tr}} \kappa^2 \langle v_{i,k} y_k, \tilde{p} \rangle$$

$$\langle \nabla u_i^*, \nabla \tilde{u} \rangle - \kappa^2 \langle m u_i^*, \tilde{u} \rangle - \kappa^2 \langle p_i m^*, \tilde{u} \rangle + \langle B p_i^*, B \tilde{u} \rangle_{\mathbf{\Gamma}_{\text{noise}}^{-1}} = -\frac{2}{n_{tr}} \sum_{k=1}^{n_{tr}} \kappa^2 \langle y_k q_{i,k}, \tilde{u} \rangle$$

$$\langle m^*, \tilde{m} \rangle_{\mathbb{E}} - \frac{1}{N_w} \sum_{i=1}^{N_w} \kappa^2 \left[\langle u_i u_i^*, \tilde{m} \rangle + \langle p_i^* p_i, \tilde{m} \rangle \right] = -\frac{2}{n_{tr} N_w} \sum_{k=1}^{n_{tr}} \sum_{i=1}^{N_w} \kappa^2 \langle v_{i,k} q_{ik}, \tilde{m} \rangle.$$

This system of equations should be compared to the system of equations for the Hessian (2.19). From this, it should be clear that the computation of m^* corresponds to the solution of another Hessian system with a right-hand side depending on the state and adjoint variables, $\{u_i\}$ and $\{p_i\}$, the incremental state and adjoint variables, $\{v_{i,k}\}$ and $\{q_{i,k}\}$, the medium parameter m, and the $\{y_k\}$. We denote this right-hand side as $\mathscr{F}$. In strong form, m^* thus solves

$$\mathcal{H}(m_{\text{MAP}}) m^* = \mathscr{F}(\{u_i\}, \{p_i\}, \{v_{i,k}\}, \{q_{i,k}\}, m, \{y_k\}).$$

Appendix B

Appendices for chapter 3

B.1 Singular values of a matrix $A \in \mathbb{R}^{2\times 2}$

Let us define the general matrix $A \in \mathbb{R}^{2\times 2}$ by

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

And let us denote its singular values decomposition as $A = U\Sigma V^T$, such that $A^TA = V\Sigma^2V^T$. And we can therefore look for the eigenvalues of the matrix A^TA given by

$$A^TA = \begin{bmatrix} a^2+c^2 & ab+cd \\ ab+cd & b^2+d^2 \end{bmatrix}.$$

We can compute these eigenvalues using the characteristic equation, $|A^TA - \lambda I| = 0$, in this case given by

$$\lambda^2 - (a^2+b^2+c^2+d^2)\lambda + (ad-bc)^2 = 0.$$

The determinant of that equation is given by

$$\begin{aligned} \Delta &= (a^2+b^2+c^2+d^2)^2 - 4(ad-bc)^2 \\ &= (a^2+c^2-b^2-d^2)^2 + 4(ab+cd)^2. \end{aligned}$$

This leads to the singular values,

$$\begin{cases} \sigma_1 = \sqrt{\dfrac{a^2+b^2+c^2+d^2+\sqrt{(a^2+c^2-b^2-d^2)^2+4(ab+cd)^2}}{2}} \\ \sigma_2 = \sqrt{\dfrac{a^2+b^2+c^2+d^2-\sqrt{(a^2+c^2-b^2-d^2)^2+4(ab+cd)^2}}{2}} \end{cases} \tag{B.1}$$

B.2 Relative medium misfits for examples in section 3.4

Table B.1: Relative medium misfits (measured in the L^2-norm) for the example in section 3.4.1.1

	m_1	m_2
independent reconstructions	23.2%	5.1%
cross-gradient	22.3%	5.2%
normalized cross-gradient	21.2%	5.0%
vectorial total variation	20.2%	5.1%
nuclear norm	20.2%	4.8%

Table B.2: Relative medium misfits (measured in the L^2-norm) for the example in section 3.4.1.2

	m_1	m_2
independent reconstructions	46.9%	5.1%
cross-gradient	46.1%	5.6%
normalized cross-gradient	46.7%	5.0%
vectorial total variation	41.1%	5.2%
nuclear norm	40.8%	5.0%

Table B.3: Relative medium misfits (measured in the L^2-norm) for the example in section 3.4.2

	α	β
independent reconstructions	2.8%	0.8%
cross-gradient	3.1%	0.7%
normalized cross-gradient	2.5%	0.4%
vectorial total variation	2.4%	0.2%
nuclear norm	2.4%	0.2%

Table B.4: Relative medium misfits (measured in the L^2-norm) for the example in section 3.4.3

	m	α
independent reconstructions	9.0%	9.9%
cross-gradient	4.9%	11.0%
normalized cross-gradient	4.9%	10.7%
vectorial total variation	8.9%	3.3%
nuclear norm	20.6%	4.5%

Appendix C

Appendix for chapter 4

C.1 Relative medium misfits for examples in section 4.3

Table C.1: Relative medium misfits (in the L^2-norm) at the minimizer for the joint inverse problem in section 4.3.1.1

ε	m_1	m_2
10	21.7%	6.6%
10^{-1}	21.1%	5.3%
10^{-3}	20.9%	5.2%
10^{-5}	20.9%	5.2%

Table C.2: Relative medium misfits (in the L^2-norm) at the minimizer for the joint inverse problem in section 4.3.1.2

ε	α	β
10^{-1}	4.6%	0.5%
10^{-3}	2.5%	0.2%
10^{-5}	2.1%	0.1%

Table C.3: Relative medium misfits (in the L^2-norm) at the minimizer for the joint inverse problem in section 4.3.3

# joint	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}
1	18.0%	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
2	14.9%	14.1%	–	–	–	–	–	–	–	–	–	–	–	–	–	–
3	13.8%	13.3%	14.2%	–	–	–	–	–	–	–	–	–	–	–	–	–
4	13.1%	11.9%	13.8%	13.2%	–	–	–	–	–	–	–	–	–	–	–	–
5	12.9%	11.7%	12.7%	12.5%	8.9%	–	–	–	–	–	–	–	–	–	–	–
6	12.7%	11.4%	12.3%	11.5%	8.6%	8.6%	–	–	–	–	–	–	–	–	–	–
7	12.3%	11.2%	12.0%	11.0%	8.4%	8.1%	9.1%	–	–	–	–	–	–	–	–	–
8	12.0%	10.9%	11.7%	10.6%	7.8%	8.0%	8.7%	7.7%	–	–	–	–	–	–	–	–
9	12.0%	10.9%	11.5%	10.7%	7.7%	8.1%	8.8%	7.7%	8.4%	–	–	–	–	–	–	–
10	11.9%	10.7%	11.5%	10.4%	7.6%	7.7%	8.9%	7.9%	8.4%	9.3%	–	–	–	–	–	–
11	11.9%	10.6%	11.3%	10.5%	7.8%	7.5%	8.6%	8.0%	8.3%	9.2%	8.5%	–	–	–	–	–
12	11.9%	10.5%	11.3%	10.6%	7.9%	7.6%	8.4%	7.8%	8.5%	9.2%	8.5%	8.0%	–	–	–	–
13	11.7%	10.4%	11.0%	10.2%	7.4%	7.2%	8.3%	7.6%	8.2%	8.8%	8.3%	7.6%	8.5%	–	–	–
14	11.7%	10.4%	10.9%	10.1%	7.4%	7.0%	8.2%	7.6%	8.1%	8.8%	8.2%	7.6%	8.5%	7.8%	–	–
15	11.5%	10.4%	10.8%	10.0%	7.3%	6.9%	7.9%	7.4%	8.1%	8.5%	8.1%	7.5%	8.3%	7.9%	8.0%	–
16	11.1%	10.1%	10.6%	9.5%	6.7%	6.7%	7.6%	6.8%	7.7%	8.2%	7.7%	7.1%	7.9%	7.5%	7.8%	7.6%

Bibliography

[1] A Abubakar, G Gao, Tarek M Habashy, and J Liu. Joint inversion approaches for geophysical electromagnetic and elastic full-waveform data. *Inverse Problems*, 28(5), 2012.

[2] Keiiti Aki and Paul G. Richards. *Quantitative Seismology*. University Science Books, 2nd edition, 2002.

[3] Alen Alexanderian, Noemi Petra, Georg Stadler, and Omar Ghattas. A-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems with regularized ℓ_0-sparsification. *SIAM Journal on Scientific Computing*, 36(5):A2122–A2148, 2014.

[4] Alen Alexanderian, Noemi Petra, Georg Stadler, and Omar Ghattas. A fast and scalable method for A-optimal design of experiments for infinite-dimensional Bayesian nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 38(1):A243–A272, 2016.

[5] Theodore W. Anderson and Michael A. Stephens. Tests for randomness of directions against equatorial and bimodal alternatives. *Biometrika*, 59(3):613–621, 1972.

[6] Uri M. Ascher, Eldad Haber, and Hui Huang. On effective methods for implicit piecewise smooth surface recovery. *SIAM Journal on Scientific Computing*, 28(1):339–358, 2006.

[7] Anthony C. Atkinson and Alexander N. Donev. *Optimum Experimental Designs*. Oxford, 1992.

[8] Haim Avron and Sivan Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)*, 58(2):17, April 2011.

[9] Biondo Biondi. *3D Seismic Imaging*. Investigations in geophysics. Society of Exploration Geophysicists, 2006.

[10] George Biros and Omar Ghattas. Parallel Lagrange–Newton–Krylov–Schur methods for PDE–constrained optimization. Part I: The Krylov–Schur solver. *SIAM Journal on Scientific Computing*, 27(2):687–713, 2005.

[11] George Biros and Omar Ghattas. Parallel Lagrange–Newton–Krylov–Schur methods for PDE–constrained optimization. Part II: The Lagrange–Newton solver and its application to optimal control of steady viscous flows. *SIAM Journal on Scientific Computing*, 27(2):714–739, 2005.

[12] Peter Blomgren and Tony F. Chan. Color TV: Total variation methods for restoration of vector-valued images. *IEEE Transactions on Image Processing*, 7(3):304–309, 1998.

[13] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[14] Xavier Bresson and Tony F Chan. Fast dual minimization of the vectorial total variation norm and applications to color image processing. *Inverse problems and imaging*, 2(4):455–484, 2008.

[15] Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg Stadler. A computational framework for infinite-dimensional Bayesian inverse problems Part I: The linearized case, with application to global seismic inversion. *SIAM Journal on Scientific Computing*, 35(6):A2494–A2523, 2013.

[16] Carey Bunks, Fatimetou M. Saleck, S. Zaleski, and Guy Chavent. Multiscale seismic waveform inversion. *Geophysics*, 60(5):1457–1473, September 1995.

[17] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

[18] Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9(263-340):227, 2010.

[19] T.F. Chan, G.H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.

[20] Tony Chan and Jianhong Shen. *Image Processing and Analysis*. Society for Industrial and Applied Mathematics, 2005.

[21] Guy Chavent. *Nonlinear Least Squares for Inverse Problems*. Springer, 2009.

[22] Benjamin Crestel, Alen Alexanderian, Georg Stadler, and Omar Ghattas. A-optimal encoding weights for nonlinear inverse problems, with application to the Helmholtz inverse problem. *Inverse Problems*, 33(7):074008, 2017.

[23] Benjamin Crestel, Georg Stadler, and Omar Ghattas. A comparative study of regularizations for joint inverse problems. *In preparation*, 2017.

[24] Benjamin Crestel, Georg Stadler, and Omar Ghattas. A primal-dual Newton method for the solution of joint inverse problems regularized with vectorial total variation. *In preparation*, 2017.

[25] Masoumeh Dashti and Andrew M. Stuart. The Bayesian approach to inverse problems. In Roger Ghanem, David Higdon, and Houman Owhadi, editors, *Handbook of Uncertainty Quantification*. Spinger, 2015.

[26] Juan Carlos De Los Reyes. *Numerical PDE-constrained optimization*. Springer, 2015.

[27] Ron S. Dembo, Stanley C. Eisenstat, and Trond Steihaug. Inexact Newton methods. *SIAM Journal on Numerical Analysis.*, 19:400–408, 1982.

[28] Peter Deuflhard. *Newton methods for nonlinear problems*, volume 35 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2004.

[29] Yiqiu Dong, Michael Hintermüller, and M. Monserrat Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *Journal of Mathematical Imaging and Vision*, 40(1):82–104, 2011.

[30] Göran Ekström, Meredith Nettles, and Adam M. Dziewoński. The global CMT project 2004-2010: Centroid-moment tensors for 13,017 earthquakes. *Physics of the Earth and Planetary Interiors*, 200-201(0):1 – 9, 2012.

[31] Bjorn Engquist and Andrew Majda. Absorbing boundary conditions for the numerical simulation of waves. *Mathematics of Computation*, 31(139):629–651, 1977.

[32] Ioannis Epanomeritakis, Volkan Akçelik, Omar Ghattas, and Jacobo Bielak. A Newton-CG method for large-scale three-dimensional elastic full-waveform seismic inversion. *Inverse Problems*, 24(3):034015 (26pp), 2008.

[33] Lawrence C. Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, second edition, 2010.

[34] Carlos Fernandez-Granda. Optimization-based data analysis. *Lecture notes at NYU-CIMS*, April 2016.

[35] Andreas Fichtner. *Full seismic waveform modelling and inversion*. Springer, 2011.

[36] Pearl H. Flath, Lucas C. Wilcox, Volkan Akçelik, Judy Hill, Bart van Bloemen Waanders, and Omar Ghattas. Fast algorithms for Bayesian uncertainty quantification in large-scale linear inverse problems based on low-rank partial Hessian approximations. *SIAM Journal on Scientific Computing*, 33(1):407–432, 2011.

[37] Luis A. Gallardo and Max A. Meju. Characterization of heterogeneous near-surface materials by joint 2D inversion of DC resistivity and seismic data. *Geophysical Research Letters*, 30(13), 2003.

[38] Luis A. Gallardo and Max A. Meju. Structure-coupled multiphysics imaging in geophysical sciences. *Reviews of Geophysics*, 49(1), 2011.

[39] Tom Goldstein and Stanley Osher. The split Bregman method for L1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[40] A. Griewank. Rates of convergence for secant methods on nonlinear problems in Hilbert space. In J. P. Hennart, editor, *Numerical Analysis, Proceedings Guanajuato , Mexico 1984*, number 1230 in Lecture Notes in Mathematics, pages 138–157, Heidelberg, 1986. Springer.

[41] A. Griewank. The local convergence of Broyden-like methods on Lipschitzian problems in Hilbert space. *SIAM Journal on Numerical Analysis*, 24:684–705, 1987.

[42] Eldad Haber. *Computational Methods in Geophysical Electromagnetics*. Mathematics in Industry. Society for Industrial and Applied Mathematics, 2014.

[43] Eldad Haber, Matthias Chung, and Felix J Herrmann. An effective method for parameter estimation with PDE constraints with multiple right hand sides. *SIAM Journal on Optimization*, 22 (3):739–757, 2012.

[44] Eldad Haber and Michal Holtzman Gazit. Model fusion and joint inversion. *Surveys in Geophysics*, 34(5):675–695, 2013.

[45] Eldad Haber, Lior Horesh, and Luis Tenorio. Numerical methods for experimental design of large-scale linear ill-posed inverse problems. *Inverse Problems*, 24(055012):125–137, 2008.

[46] Eldad Haber, Lior Horesh, and Luis Tenorio. Numerical methods for the design of large-scale nonlinear discrete ill-posed inverse problems. *Inverse Problems*, 26(2):025002, 2010.

[47] Eldad Haber and Jan Modersitzki. Intensity gradient based registration and fusion of multi-modal images. *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2006*, pages 726–733, 2006.

[48] Eldad Haber, Kees van den Doel, and Lior Horesh. Optimal design of simultaneous source encoding. *Inverse Problems in Science and Engineering*, pages 1–18, 2014.

[49] Per Christian Hansen and Dianne Prost O'Leary. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM Journal on Scientific Computing*, 14(6):1487–1503, 1993.

[50] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total variation-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28(1):1–23, 2006.

[51] Michael Hinze, Rene Pinnau, Michael Ulbrich, and Stefan Ulbrich. *Optimization with PDE Constraints*. Springer, 2009.

[52] Bamdad Hosseini and Nilima Nigam. Well-posed Bayesian inverse problems: beyond Gaussian priors. *arXiv preprint arXiv:1604.02575*, 2016.

[53] Thomas J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover, New York, 2000.

[54] Michael F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 19(2):433–450, 1990.

[55] Tobin Isaac, Noemi Petra, Georg Stadler, and Omar Ghattas. Scalable and efficient algorithms for the propagation of uncertainty from data through inference to prediction for large-scale problems, with application to flow of the Antarctic ice sheet. *Journal of Computational Physics*, 296:348–368, September 2015.

[56] Khalid Jalalzai. Some remarks on the staircasing phenomenon in total variation-based image denoising. *Journal of Mathematical Imaging and Vision*, 54(2):256–268, 2016.

[57] Brian L. N. Kennett. *The Seismic Wavefield: Volume 1, Introduction and Theoretical Development*. Cambridge University Press, 2001.

[58] Florian Knoll, Martin Holler, Thomas Koesters, Ricardo Otazo, Kristian Bredies, and Daniel K Sodickson. Joint MR-PET reconstruction using a multi-channel image regularizer. *IEEE transactions on medical imaging*, 36(1):1–16, 2017.

[59] Jerome R. Krebs, John E. Anderson, David Hinkley, Ramesh Neelamani, Sunwoong Lee, Anatoly Baumstein, and Martin-Daniel Lacasse. Fast full-wavefield seismic inversion using encoded sources. *Geophysics*, 74(6):WCC177–WCC188, 2009.

[60] Maokun Li, Lin Liang, Aria Abubakar, and Peter M van den Berg. Structural similarity regularization scheme for multiparameter seismic full

waveform inversion. In *SEG Technical Program Expanded Abstracts 2013*, pages 1089–1094. Society of Exploration Geophysicists, 2013.

[61] Anders Logg, Kent-Andre Mardal, and Garth Wells. *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS book*, volume 84. Springer Science & Business Media, 2012.

[62] Anders Logg and Garth N. Wells. DOLFIN: Automated finite element computing. *ACM Transactions on Mathematical Software*, 37(2):Article 20, 2010.

[63] Edgar Manukyan, Hansruedi Maurer, and Andre Nuber. Improvements to elastic full waveform inversion using cross-gradient constraints. In *SEG Technical Program Expanded Abstracts 2016*, pages 1506–1510. Society of Exploration Geophysicists, 2016.

[64] Gary S Martin, Robert Wiley, and Kurt J Marfurt. Marmousi2: An elastic upgrade for Marmousi. *The Leading Edge*, 25(2):156–166, 2006.

[65] James Martin, Lucas C. Wilcox, Carsten Burstedde, and Omar Ghattas. A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion. *SIAM Journal on Scientific Computing*, 34(3):A1460–A1487, 2012.

[66] Vladimir Alekseevich Morozov. On the solution of functional equations by the method of regularization. In *Soviet Math. Dokl*, volume 7, pages 414–417, 1966.

[67] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Verlag, Berlin, Heidelberg, New York, second edition, 2006.

[68] Théodore Papadopoulo and Manolis I. A. Lourakis. *Estimating the Jacobian of the Singular Value Decomposition: Theory and Applications*, pages 554–570. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.

[69] Noemi Petra and Georg Stadler. Model variational inverse problems governed by partial differential equations. Technical Report 11-05, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, 2011.

[70] R. Gerhard Pratt, C. Shin, and G. J. Hicks. Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion. *Geophysical Journal International*, 133(2):341–362, 1998.

[71] Friedrich Pukelsheim. *Optimal Design of Experiments*. John Wiley & Sons, New-York, 1993.

[72] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.

[73] Partha S Routh, Sunwoong Lee, Ramesh Neelamani, Jerome R Krebs, Spyridon Lazaratos, and Carey Marcinkovich. Simultaneous source encoding and source separation as a practical solution for full wavefield inversion, September 9 2011. US Patent App. 13/229,252.

[74] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.

[75] Oguz Semerci, Guangdong Pan, Maokun Li, Lin Liang, and Tarek Habashy. Joint electromagnetic and seismic inversion for petrophysical parameters using multi-objective optimization. In *SEG Annual Meeting*, Denver, CO, 26-31 October 2014. SEG.

[76] Alexander Shapiro, Darinka Dentcheva, and Andrezj Ruszczynski. *Lectures on Stochastic Programming: Modeling and Theory*. Society for Industrial and Applied Mathematics, 2009.

[77] Klara Steklova and Eldad Haber. Joint hydrogeophysical inversion: state estimation for seawater intrusion models in 3D. *Computational Geosciences*, 1(21):75–94, 2016.

[78] Andrew M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[79] William W. Symes. Source synthesis for waveform inversion. Technical report, Rice University, CAM report TR10-12, 2010.

[80] Lonny L Thompson and Peter M Pinsky. Acoustics. *Encyclopedia of computational mechanics*, 2004.

[81] Loyd N. Trefethen and David Bau. *Numerical Linear Algebra*. SIAM, Philadelphia, 1997.

[82] Fredi Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, volume 112 of *Graduate Studies in Mathematics*. American Mathematical Society, 2010.

[83] Dariusz Uciński. *Optimal measurement methods for distributed parameter system identification*. CRC Press, Boca Raton, 2005.

[84] Umberto Villa, Noemi Petra, and Omar Ghattas. hIPPYlib: An Extensible Software Framework for Large-Scale Deterministic and Linearized Bayesian Inverse Problems. To be submitted, 2016.

[85] Jean Virieux and Stephane Operto. An overview of full-waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26, 2009.

[86] Curt R. Vogel. *Computational Methods for Inverse Problems*. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[87] Dafang Wang, Robert M. Kirby, Rob S. MacLeod, and Chris R. Johnson. Inverse electrocardiographic source localization of ischemia: An optimization framework and finite element solution. *Journal of Computational Physics*, 250:403 – 424, 2013.

[88] Lucas C. Wilcox, Georg Stadler, Tan Bui-Thanh, and Omar Ghattas. Discretely exact derivatives for hyperbolic PDE-constrained optimization problems discretized by the discontinuous Galerkin method. *Journal of Scientific Computing*, 63(1):138–162, 2015.

[89] Changxi Zhou, Wenying Cai, Yi Luo, Gerard T. Schuster, and Sia Hassanzadeh. Acoustic wave-equation traveltime and waveform inversion of crosshole seismic data. *GEOPHYSICS*, 60(3):765–773, 1995.

[90] Hejun Zhu, Ebru Bozdağ, Daniel Peter, and Jeroen Tromp. Structure of the European upper mantle revealed by adjoint tomography. *Nature Geoscience*, 5:493–498, 2012.

Vita

Benjamin (Ben) Crestel was born in France where he obtained his "Baccalauréat" in Mathematics in 1999 summa cum laude ("mention très bien"). He then joined the Engineering Curriculum attending the classical two years of Preparatory Classes, majoring in Mathematics and Physics. Following these two years, he took the National Examination for Engineering Curriculum and was accepted the first time at the Ecole Nationale des Ponts et Chaussées (ENPC, Paris). He specialized in earthquake engineering and the behavior of large bridges. He completed his undergraduate training with an exchange at Ecole Polytechnique de Montréal where he focused on the probabilistic treatment of seismic signals. He worked for 2 years as a bridge engineer for Buckland & Taylor, Ltd in Vancouver, BC before deciding to go back to academia. He completed a Master of Science in Applied Mathematics at Simon Fraser University in Vancouver, BC in 2011 under the supervision of Dr. Robert Russell and Dr. Steven Ruuth. In his thesis, Ben developed a technique to compute adaptive moving mesh on general surfaces. He then joined the Computational Science, Engineering and Mathematics Ph.D. program at the Institute of Computational Science, Engineering and Mathematics in the Spring 2012.