Abstract. The Alternating Direction Method of Multipliers (ADMM) provides a natural way of solving inverse problems with multiple partial differential equations (PDE) forward models and nonsmooth regularization. ADMM allows splitting these large-scale inverse problems into smaller, simpler sub-problems, for which computationally efficient solvers are available. In particular, we apply large-scale second-order optimization methods to solve the fully-decoupled Tikhonov regularized inverse problems stemming from each PDE forward model. We use fast proximal methods to handle the nonsmooth regularization term. In this work, we discuss several adaptations (such as the choice of the consensus norm) needed to maintain consistency with the underlining infinite-dimensional problem. We present two imaging applications inspired by electrical impedance tomography and quantitative photoacoustic tomography to demonstrate the proposed method’s effectiveness.

Key words. Inverse problems, PDE constrained optimization, non smooth optimization, electrical impedance tomography, photoacoustic tomography

AMS subject classifications. 35Q62, 62F15, 35R30, 35Q93, 65C60, 65K10, 49M15, 49M37, 1. Introduction. Partial differential equations (PDEs) are used in various fields to model complex systems adhering to physical principles. However, PDEs often depend on unknown or uncertain parameters that cannot be measured directly [8], such cases can be framed as a type of inverse problem. An inverse problem has the goal of estimating a parameter field given a set of possibly noisy data. This data is related to the parameter via a parameter-to-observable map. We will focus on this map being the solution to a PDE model. The inverse problem can then be formulated as an infinite-dimensional optimization problem with the objective being to minimize some distance between the measured data and solution to the PDE model with a regularization term on the parameter.

The main issue that separates an inverse problem from other types of optimization problems is the concept of ill-posedness [5]. Ill-posedness means that solutions to these PDE relationships can face high-sensitivity to noise or small perturbations in the data creating larger perturbations in solutions or non-uniqueness; two distinct solutions could arise from one state. To account for noise, we then add a regularization term that enforces desirable results. Similarly, the proper regularization function can be used to fix the problem of non-uniqueness so that the objective is minimized around the correct solution. The regularization is decided by choosing what properties to enforce and can thus be a broad class of functions.

The infinite-dimensional nature of inverse problems means that we have to utilize a discretization process for numerical solutions at some point in the solution process. To solve this issue we implement an optimize-then-discretize approach. This means we first derive the optimality conditions and derivatives in a continuous setting then implement a discretization process for numerical results [8]. In this work, we use the finite element method (FEM) for discretization. FEM is the process in which

* Funding: This work was funded by the National Science Foundation under Grant No ACI-1550593.
† Electrical & Systems Engineering, Washington University in St Louis, St. Louis, MO, USA
‡ Electrical & Systems Engineering, Washington University in St Louis, St. Louis, MO, USA (uvilla@wustl.edu).
a finite mesh approximates a continuous domain. Functions over the domain are approximated by the span of a set of basis functions over the mesh; usually, piecewise polynomial [10, 16, 19]. This numerical solution can be made arbitrarily accurate with a sufficiently fine mesh and a wider array of basis functions. However, this increased accuracy creates the trade-off of the problem becoming very large in scale.

In solving inverse problems of this form, we will need to apply regularization to account for noise and ill-posedness of the problem. In many cases, it is advantageous to apply nonsmooth regularization to reinforce desired behaviors and prior expectations. One such example of a useful nonsmooth regularization is total variation. Total variation has similar behaviors to regularization on the $L^2$ norm of the gradient but places a higher penalty on smaller values and a lower penalty on larger values of the gradient. Total variation allows the reconstructed parameter to vary more but still be primarily piecewise constant[18]. However, using nonsmooth regularization is problematic because the methods that best solve PDE constrained optimization problems, INCG, require well-defined derivatives.

One method of solving problems of this form involves the Alternating Direction Method of Multipliers(ADMM). The ADMM is a proximal point algorithm that is well suited for distributed convex optimization. The method was first formulated in the 1970s with works such as [17] and splits large scale problems into multiple convex subproblems. Its potential for a variety of large scale problems, including machine learning, has been widely detailed in such works as [4], which also provides a rigorous analysis of the ADMM. The authors in [23] further demonstrated that ADMM is well suited in a general optimization setting and is globally convergent for nonsmooth and nonconvex problems. With the development of ADMM focusing on large-scale problems and nonsmooth regularization, it is natural to have applications to image reconstruction and inverse problems governed by PDE forward models. The first application of ADMM for optimization involving PDEs was demonstrated in [12] and applied to solve several continuum mechanics problems. In [22, 1] ADMM is applied to classical image reconstruction problems with a linear imaging operator, such as blurring and denoising operators, with total variation regularization. In [11] this methodology is further developed to explore inverse problems with PDE constraints with linear forward models and a version of ADMM with a weighted consensus relationship to increase progress made in early iterations. Similarly, [25] implements ADMM for solving an inverse problem related to seismology and demonstrates the compatibility of ADMM with nonsmooth regularization terms, including $L^1$ regularization and second-order total variation.

This paper presents an application of the alternating direction method of multipliers (ADMM) for solving inverse problems governed by multiple PDE forward problems and nonsmooth regularization. We demonstrate ADMM’s natural ability to handle multiple PDE forward problems and nonsmooth regularization functionals by splitting this large-scale problem into subproblems for which efficient solution methods are available. The specific contribution of our work is the following. We demonstrate the effectiveness of ADMM for solving inverse problems governed by multiple PDE models. In particular, we will demonstrate this framework with PDE relationships leading to nonlinear parameter to observable maps. Our framework will also utilize a consensus ADMM equipped with an arbitrary inner product for equality enforcement. We will also demonstrate how using different inner products and norms can lead to numerical stability for solving variational problems. Throughout, we will quantitatively demonstrate how ADMM reduces computational costs for large-scale problems compared to traditional methods while still achieving sufficient accuracy.
The remainder of the paper is structured as follows. Section 2 provides a brief overview of ADMM. In particular, we recall the scaled formulation of the ADMM algorithms, adaptive weights, and robust stopping criteria. In Section 4, we provide a theoretical basis for PDE-constrained inverse problems and how the scaled-ADMM can naturally be applied to such problems. In Section 3 we provide a general outline of deterministic inverse problems and a general optimization method, inexact Newton Conjugate Gradient, for inverse problems. In Section 4, we will also introduce two algorithms for applying ADMM to large inverse problems. In Section 5, we look at a model problem related to electrical impedance tomography and perform numerical studies to show the novelty and merit in using ADMM for various problem sizes with multiple PDE models. In these numerical studies, we also show the effect of implementing a modified consensus norm, solving for local inversion parameters inexactly. In Section 6 we demonstrate the application of ADMM to an inverse problem found in quantitative photoacoustic tomography.

2. The Alternating Direction Method of Multipliers (ADMM). The ADMM is an algorithm for solving large scale constrained optimization problems whose objective involves the sum of a well-behaved twice differential term and another term that may be nonsmooth. The ADMM splits this large problem into separate smaller subproblems, one of which will involve the nonsmooth objective term. These subproblems are solved iteratively, and equality between their solutions is reinforced with a consensus term.

In this section, we recall the various formulation of ADMM and some commonly used heuristic to accelerate ADMM convergence and robust stopping criteria. The ADMM is part of a class of algorithms, called proximal point algorithms [17], which require very little to guarantee convergence.

We consider the following minimization problem

\begin{equation}
\min_{m \in \mathcal{M}} \mathcal{J}(m) := \frac{1}{q} \sum_{i=1}^{q} \mathcal{L}_i(m) + \mathcal{R}(m),
\end{equation}

where $m$ is the sought after parameter in possibly infinite dimensional Hilbert space $\mathcal{M}$, and $q > 0$. The functionals $\mathcal{L}_i : \mathcal{M} \mapsto \mathbb{R}$ are assumed to be smooth (twice differentiable) and expensive to evaluate. The functional $\mathcal{R} : \mathcal{M} \mapsto \mathbb{R}$ is assumed convex and non-smooth. Furthermore, we assume that computationally efficient methods are available to solve proximal problems stemming from $\mathcal{R}$.

For ease of notation, we assume $q = 1$ in the presentation of the ADMM algorithm below. The general case $q \geq 1$ is presented in Section 4 in the contest of inverse problems governed by partial differential equation forward models.

The ADMM introduces a new variable $z \in \mathcal{M}$ and changes the optimization problem 2.1 to the form (2.2).

\begin{equation}
\min_{m, z \in \mathcal{M}, \ s.t. \ m - z = 0} \mathcal{L}(m) + \mathcal{R}(z),
\end{equation}

For a general equality constrained optimization problem given by (2.2) we first form the augmented Lagrangian for some $\rho > 0$ by (2.3).

\begin{equation}
\mathcal{L}_\rho(m, z, y) = \mathcal{L}(m) + \mathcal{R}(z) + \langle y, m - z \rangle + \frac{\rho}{2} ||m - z||^2
\end{equation}
Where $y \in \mathcal{M}$ is a Lagrange multiplier for the constraint $m = z$. Here $\langle \cdot, \cdot \rangle$ denotes the inner product and $\| \cdot \|$ is the norm induced by the inner product.

The augmented Lagrangian has a few noteworthy properties. First, if the Lagrangian has a unique stationary point, then the augmented Lagrangian will have the same unique stationary point \cite{2}. This invariance on the stationary point means optimizing using the augmented Lagrangian will result in the same outcome as the regular Lagrangian, and a choice of $\rho$ does not affect the outcome.

Second, the added quadratic term for penalizing the constraint ensures convergence of the ADMM iterative process \cite{23}. The augmented Lagrangian is also more desirable than other penalization or barrier functions because it will create well-behaved steps in the iterative process \cite{2}.

Third, the lack of requirements on $\rho$ means that one’s choice of $\rho$ can tuned to accelerate the convergence of the ADMM. A larger $\rho$ will make the variables $m$ and $z$ more accurately agree at every iteration. A smaller $\rho$ will make the $m$ and $z$ quickly approximate the optimal values of each subproblem but lack complete agreement. Choosing the optimal $\rho$ is then a trade-off between these two qualities. Algorithm 1 summarizes the ADMM.

| Algorithm 1: The General ADMM |
|-------------------------------|
| Begin with starting points $(m^0, z^0, y^0)$ |
| while While convergence criterion is not met do |
| $m^{k+1} = \text{argmin}_m L_\rho(m, z^k, y^k)$ |
| $z^{k+1} = \text{argmin}_z R(z) + \rho \|m^{k+1} - z + y^k\|^2$ |
| $y^{k+1} = y^k + \rho(m^{k+1} - z^{k+1})$ |
| end |

If we scale $y$ by a factor of $\frac{1}{\rho}$ and do a least squares completion then we can transform the ADMM to the scaled ADMM given by Algorithm 2\cite{4}. This form is useful because the problem is now only in terms of the objective functions and quadratic term. Furthermore, in the scaled form of ADMM the update of $y$ is independent of $\rho$.

| Algorithm 2: The Scaled ADMM |
|-------------------------------|
| Begin with starting points $(m^0, z^0, y^0)$ |
| while While convergence criterion is not met do |
| $m^{k+1} = \text{argmin}_m L(m) + \rho \|m - z^k + y^k\|^2$ |
| $z^{k+1} = \text{argmin}_z R(z) + \rho \|m^{k+1} - z + y^k\|^2$ |
| $y^{k+1} = y^k + m^{k+1} - z^{k+1}$ |
| end |

If the Lagrangian $L_0$ has a unique saddle point at $(m^*, z^*, y^*)$ and $L$ and $R$ are proper closed and convex functions then as $k \to \infty$ then $(m^k, z^k, y^k) \to (m^*, z^*, y^*)$ \cite{12}. That being said, ADMM will only reach moderate accuracy in a few iterations and requires many following iterations for high-precision convergence\cite{4, 11}. Luckily in many applications, including those considered here, this is sufficient. The ADMM is so useful for large-scale problems because it splits these problems into multiple sub-problems updated iteratively. This splitting means we only require the resources to solve these smaller problems, reducing memory requirements, problem complexity, and processing power. The ADMM is designed to be entirely parallelizable and only requires communication to the consensus from each instance variable once per update. Therefore it requires significantly less communication time compared with other parallelized solutions \cite{25}.
2.1. Adaptive $\rho$ choice. Since the saddle point of (2.3) is independent of $\rho$, one can implement an adaptive choice of $\rho$ depending on the disagreement between the parameters and the rate at which they change.

To make this choice, two types of residuals are used as a measure of convergence following the framework outline in [24]. We denote the primal residual and the dual residual at the $k$-th iteration respectively as

$$r^k = m^k - z^k, \quad s^k = \rho(z^k - z^{k-1}).$$

The primal residual serves as a measure for the agreement between $m$ and $z$. The primal residual being zero implies that the system has reached primal feasibility. Meanwhile, the dual residual serves as a measure for the settling of $z$ and it being zero indicates the system has reached dual feasibility [24]. The primal residual being much greater than the dual residual means too much weight is given to the objective function over the agreement of $m$ and $z$. This issue is fixed by increasing the value of $\rho$. Similarly, if the dual residual is much greater than the primal residual, there is not enough weight on the consensus; thus, $\rho$ should be decreased.

To speed the convergence of this process the following heuristic can be used. For a fixed $\rho^k$ following the calculations of $m^k, z^k, r^k$ and $s^k$ we can make the following choice of $\rho^{k+1}$ by

$$\rho^{k+1} = \begin{cases} \tau \rho^k & \|r^k\| > \mu \|s^k\| \\ \rho^k/\tau & \|s^k\| > \mu \|r^k\| \\ \rho^k & \text{otherwise} \end{cases} \tag{2.4}$$

Where $\mu, \tau > 1$ are tunable parameters. Commonly these are chosen to by $\mu = 10$ and $\tau = 2$.

2.2. Stopping criterion. Using the residuals we can also implement a gradient free stopping criterion adapted from [24, 4]. To do this we simply choose a small absolute tolerance $\varepsilon_a > 0$ and a small relative tolerance $\varepsilon_r > 0$. We then stop the iterative process at the first iteration when the following criterion is met.

$$\|r^k\| \leq \varepsilon_{abs} + \varepsilon_{rel}\|m^k\| \quad \text{and} \quad \|s^k\| \leq \varepsilon_{abs} + \varepsilon_{rel}\|z^k\|$$

It should be noted that $\|\cdot\|$ is the norm for $\mathcal{M}$.

3. Inexact Newton conjugate gradient method for PDE-constrained inverse problems. This section presents a brief outline of the formulation of deterministic inverse problems and a specific method for their solution. We begin by outlining the notation that we will use relating to inverse problems and their formalism. Then we explain how to apply the Inexact Newton Conjugate Gradient (INCG) method to solve problems of this form.

3.1. Deterministic inverse problems. An inverse problem has the goal of reconstructing for a parameter $m$ given a measurement $d \in \mathcal{D}$. Mathematically the forward model of an inverse problem can be expressed as (3.1).

$$(3.1) \quad d = \mathcal{F}(m) + e,$$

where $\mathcal{F} : \mathcal{M} \to \mathcal{D}$ is a map from the parameter space to the observation space, and $e$ is random noise present in measurements. The forward map $\mathcal{F}$ is often a
relationship that is not directly available; for example, we will be focusing on implicit PDE relationships. For direct evaluations we can then introduce a state variable $u \in \mathcal{U}$ satisfying some relationship $r(m, u) = 0$. The goal of this transformation is to create an explicit relationship $d = \mathcal{B}(u)$, where $\mathcal{B}: \mathcal{U} \rightarrow \mathcal{D}$ is an observation operator.

Given $d$, our goal is to reconstruct for $m$ while adjusting for the presence of the noise $e$. This can be characterized as minimizing the cost functional

$$J(m) := L(m) + R(m)$$

where $L(m) = \frac{1}{2} \| \mathcal{F}(m) - d \|^2$.

Here $R$ is a regularization function and $L$ is a data fidelity term that is minimized when $\mathcal{F}(m) = d$. Substituting in $\mathcal{F}(m) = \mathcal{B}(u)$, the new goal of the inverse problem is to solve the minimization problem in

$$\min_{m \in \mathcal{M}, u \in \mathcal{U}} J(m) = \frac{1}{2} \| \mathcal{B}(u) - d \|^2 + \mathcal{R}(m)$$

s.t. $r(m, u) = 0$

With this problem now formulated as a constrained optimization problem we can express the Lagrangian in (3.4) in terms of $m$, $u$ and an adjoint variable $p \in \mathcal{M}$.

$$\mathcal{L}(m, u, p) = \frac{1}{2} \| \mathcal{B}(u) - d \|^2 + \langle p, r(m, u) \rangle.$$  

With this Lagrangian formalism expressed, we can derive the infinite-dimensional analogs of the gradient and Hessian. Letting subscripts denote Gâteaux derivative, we can denote the gradient of the cost functional (3.2) in a a direction $\tilde{m} \in \mathcal{M}$ at a point $m_0$ as

$$(\mathcal{G}(m_0), \tilde{m}) = (\mathcal{R}_m(m_0), \tilde{m}) + \langle p_0, r_m(u_0, m_0) [\tilde{m}] \rangle.$$  

Above $u_0$ is the solution of the forward problem

$$\langle \hat{p}, r(u_0, m_0) \rangle = 0, \forall \hat{p},$$  

which is obtained by requiring variations of (3.4) to vanish for all directions $\hat{p}$. $p_0$ is the solution to the adjoint problem

$$\langle p_0, r_u(u_0, m_0) [\hat{u}] \rangle + \langle u(\mathcal{B}(u_0) - d), \mathcal{B} \hat{u} \rangle = 0, \forall \hat{u},$$  

which is obtained by requiring variations of (3.4) to vanish for all directions $\hat{u} \in \mathcal{U}$. Similarly, to derive the Hessian action, we consider the second-order Lagrangian

$$\mathcal{L}^H(m, u, p; \hat{m}, \hat{u}, \hat{p}) = (\mathcal{G}(m), \hat{m})$$  

$$+ \langle \hat{p}, r(u, m) \rangle$$  

$$+ \langle p, r_u(u, m) [\hat{u}] \rangle + \langle \mathcal{B}(\hat{u}), \mathcal{B}(u) - d \rangle.$$  

The Hessian in a direction $\hat{m} \in \mathcal{M}$ evaluated at $m = m_0$ is then the Gâteaux derivative of $\mathcal{L}^H$ with respect to $m$ and given as
(3.9) \( (\tilde{m}, \mathcal{H}(m_0)\tilde{n}) = (\tilde{m}, \mathcal{R}_{mm}(m_0)[\tilde{m}]) + (p_0, r_{mm}(u_0,m_0)[\tilde{m}, \tilde{m}]) + (\tilde{p}, r_m(u_0,m_0)[\tilde{m}]) + (p_0, r_{um}(u_0,m_0)[\tilde{u}, \tilde{m}]), \quad \forall \tilde{m} \in \mathcal{M} \)

Where incremental state \( \tilde{u} \) and incremental adjoint \( \tilde{p} \) solve the so-called incremental forward and incremental adjoint problems, which are obtained by setting to zero variations of (3.8) with respect to \( p \) and \( u \) respectively. Optimality conditions still hold in the infinite dimensional setting. This means that the cost functional in (3.2) and (3.3) is minimized at a point when the gradient (3.5) is identically zero for every \( \tilde{m} \) and the Hessian is positive definite.

3.2. Inexact Newton Conjugate Gradient(INCG) for solving inverse problems. With the infinite-dimensional derivatives derived in Section 3.1 we can apply traditional minimization algorithms, including gradient descent and Newton descent. Here we will outline the application of Inexact Newton Conjugate Gradient(INCG) for solving an inverse problem. Now we proceed with discretizing the problem for a numerical solution. It is then necessary to note that the gradient, Hessian, and optimality conditions are evaluated as discretized statements of variational problems instead of first discretizing the problem and then treating it as a finite-dimensional optimization problem. The INCG algorithm is shown in Algorithm 3.

Here (3.10) is known as the Eisenstat-Walker condition and results in desirably fast local convergence [9]. This condition leads to superlinear convergence of Algorithm 3 while at the same time drastically reducing the number of necessary iterations to solve the Newton system.

4. Application of ADMM to the solution of inverse problems governed by multiple PDE constraints. In this section, we consider the minimization problem (2.1), in the context of an infinite dimensional inverse problem with PDE forward problems. Then, \( m \in \mathcal{M} \) belongs to some Sobolev space defined on a domain \( \Omega \subset \mathbb{R}^d \) (\( d = 1, 2, 3 \)), the functionals \( \mathcal{L}_i(m) \) represent the smooth data fidelity terms, those evaluation involve the solution of the PDE, and \( \mathcal{R}(m) \) the regularization functional. Specifically, we consider the following form of the data-fidelity term

\[
(4.1) \quad \mathcal{L}(m) = \frac{1}{q} \sum_{i=1}^{q} \mathcal{L}_i(m) := \frac{1}{2q} \sum_{i=1}^{q} \| \mathcal{F}_i(m) - d_i \|^2,
\]

where \( d_i \in \mathcal{D} \) (\( i = 1, \ldots, q \)) represent the data, and \( \mathcal{F}_i : \mathcal{M} \mapsto \mathcal{D} \) is the parameter to observable map. For the applications we focus on \( \mathcal{F}_i \) will be the composition of a PDE solver and an observation operator [20].

4.1. ADMM with consensus equations. In Section 2, we introduce ADMM for a single data fidelity term \( \mathcal{L}(i.e. \ for \ q = 1) \). Here we can generalize this to deal with multiple PDE-based forward models in a special version of the global consensus problem [4]. If we are given \( q \) data sets \( \{d_i\}_{i=1}^{q} \) and their corresponding parameter to observable maps \( \{\mathcal{F}_i : \mathcal{M} \mapsto \mathcal{D}\}_{i=1}^{q} \), our goal would normally be to solve for \( m_{\text{MAP}} \) as in (4.2).

\[
(4.2) \quad m_{\text{MAP}} = \arg\min_m \frac{1}{2q} \sum_{i=1}^{q} \| \mathcal{F}_i(m) - d_i \|^2 + \mathcal{R}(m)
\]
Algorithm 3: The Inexact Newton Conjugate Gradient algorithm for solving inverse problems

Start with $i = 0$.
Given $m_0$ solve the forward problem (3.6) to obtain $u_0$.
Given $m_0, u_0$ compute the cost functional $J_0$ using (3.2).

while $i < \text{max}_\text{iter}$ do
    Given $m_i, u_i$ solve the adjoint problem (3.7) to obtain $p_i$
    Given $m_i, u_i, p_i$ compute the gradient $g_i$ using (3.5).
    if $\|g_i\| \leq \tau$ then
        break
    end
    Given $m_i, u_i, p_i$ define a linear operator $H_i$ implementing the Hessian action (3.9).
    Using Conjugate gradients, find a search direction $\hat{m}_i$ such that
    $$(3.10) \quad \|H_i \hat{m}_i + g_i\| \leq \eta_i \|g_i\|, \quad \eta_i = \left(\frac{\|g_i\|}{\|g_0\|}\right)^{1/2}$$
    Set $j = 0, \alpha^{(0)} = 1$ while $j < \text{max}_\text{backtracking}_\text{iter}$ do
        Set $m^{(j)} = m_i + \alpha^{(j)} \hat{m}_i$
        Given $m^{(j)}$ solve the forward problem (3.6) to obtain $u^{(j)}$
        Given $m^{(j)}$ and $u^{(j)}$ compute the cost $J^{(j)}$ using (3.2)
        if $J^{(j)} < J_i + \alpha^{(j)} \text{Carm} \sum_{i=1}^{q} \eta_i \|g_i\|$ \label{eq:1}
            $m_{i+1} \leftarrow m^{(j)}, \quad J_{i+1} \leftarrow J^{(j)}$
            break
        end
        $\alpha^{(j+1)} \leftarrow \alpha^{(j)}/2, j \leftarrow j + 1$
    end
    $i \leftarrow i + 1$
end

We can instead split this parameter for each model and data set and apply an equality constraint between its multiple instances.

$$m_1 = \ldots = m_q = z.$$ \label{eq:4}

This also requires the introduction of $q$ different multipliers $\{y_i\}_{i=1}^q \subset M$ for each of these equality relationships. $y_i$ will act as the Lagrange multiplier for the equality relationship $m_i = z$. Using this, we can form the scaled augmented Lagrangian over all these variables and change our goal to solving (4.4).

$$\min_{m_i \in M, z \in Z} \frac{1}{2q} \sum_{i=1}^{q} \|F_i(m_i) - d_i\|^2 + R(z)$$

s.t. $m_i - z = 0$ for $i = 1, \ldots, q$

If we scale $\rho$ by a factor of $\frac{1}{q}$ then this problem will result in the augmented
Lagrangian given by (4.5).

\[ L_\rho(\{m_i\}_{i=1}^q, z, \{y_i\}_{i=1}^q) = \frac{1}{2q} \sum_{i=1}^q \| F_i(m_i) - d_i \|^2 + R(z) + \frac{1}{q} \sum_{i=1}^q \langle y_i, m_i - z \rangle + \frac{\rho}{2q} \sum_{i=1}^q \| m_i - z \|^2 \]  

(4.5)

With this form the optimal argument for each \( m_i \) will be independent of all other parameters \( m_1, \ldots, m_{i-1}, m_{i+1}, \ldots, m_q \). This now means that at each step of the scaled ADMM each \( m_i \) can be updated in the simplified form given by (4.6).

\[ m_i^{k+1} = \text{argmin}_{m_i} \frac{1}{2q} \| F_i(m_i) - d_i \|^2 + \frac{\rho}{2q} \| m_i - z^k + y^k \|^2. \]  

(4.6)

This greatly reduces the computational complexity of each update opposed to only having one parameter variable for every model and simplifies the regularization to always being a Tikhonov regularization for the update. Similarly consensus variable \( z \) will be updated as

\[ z^{k+1} = \text{argmin} z R(z) + \frac{\rho}{2q} q \sum_{i=1}^q \| m_i - z + y_i \|^2. \]  

(4.7)

This update is entirely free of the terms related to the PDE, which means we can solve it with a broader class of optimization methods. Thus the update process can be described by Algorithm 4.

Algorithm 4: The Scaled ADMM for parameter inversion with multiple PDE's

Let \( q \) be the number of PDE relationships

Begin with starting points \((\{m_i^0\}_{i=1}^q, z^0, y^0)\)

while convergence criterion is not met, \( k = 1, \ldots \) do

for \( i = 1, \ldots, q \) do

\( m_i^{k+1} \) is updated as in (4.6)

end

\( z^{k+1} \) is updated as (4.7)

for \( i = 1, \ldots, q \) do

\( y_i^{k+1} = y_i^k + (m_i^{k+1} - z^{k+1}) \)

end

Update \( \rho^{k+1} \) following (2.4)

end

At this point, we observe that

\[
\frac{1}{q} \sum_{i=1}^q \| m_i + y_i - z \|^2 = \| z \|^2 - 2\langle z, \frac{1}{q} \sum_{i=1}^q m_i + y_i \rangle + \frac{1}{q} \sum_{i=1}^q \| m_i + y_i \|^2 = \\
\| z \|^2 - 2\langle z, \bar{m} + \bar{y} \rangle + \| \bar{m} + \bar{y} \|^2 + \frac{1}{q} \sum_{i=1}^q \| m_i + y_i \|^2 - \| \bar{m} + \bar{y} \|^2 =
\]
Algorithm 4.1 The Mean based Scaled ADMM for parameter inversion with multiple PDE’s

Let $q$ be the number of PDE relationships

Begin with starting points $(\{m_i^0\}_{i=1}^q, z^0, y^0)$

while While convergence criterion is not met, $k = 1, \ldots$
do
  for $i = 1, \ldots, q$ do
    $m_i^{k+1}$ is updated as in (4.6)
  end
  Set $\bar{m} = \frac{1}{q} \sum_{i=1}^q m_i^{k+1}$ and, $\bar{y} = \frac{1}{q} \sum_{i=1}^q y_i^{k+1}$
  $z^{k+1}$ is updated as in (4.8)
  for $i = 1, \ldots, q$ do
    $y_i^{k+1} = y_i^k + \frac{1}{q} (m_i^{k+1} - z^{k+1})$
  end
  Update $\rho^{k+1}$ following (2.4)
endo

$||\bar{m} + \bar{y} - z||^2 + \frac{1}{q} \sum_{i=1}^q ||m_i + y_i||^2 - ||\bar{m} + \bar{y}||^2$, where $\bar{m} = \frac{1}{q} \sum_{i=1}^q m_i$, $\bar{y} = \frac{1}{q} \sum_{i=1}^q y_i$

The term $\frac{1}{q} \sum_{i=1}^q ||m_i + y_i||^2 - ||\bar{m} + \bar{y}||^2$ is constant in $z$, which means that the update for $z$ is equivalent to

(4.8) $z^{k+1} = \arg\min_z \mathcal{R}(z) + \frac{\rho}{2} ||\bar{m} - \bar{y} + z||^2$.

We can then implement a mean based approach that simplifies the optimization process for updating $z$. This mean based approach is shown in Algorithm 4.1.

This process can further be expanded by splitting the parameter variables by spatially dependent subregions and implementing variable asynchronous weights for consensus update as demonstrated in [11].

5. Numerical studies: electrical impedance tomography problem. To demonstrate the effectiveness of ADMM, we consider a model problem based on electrical impedance tomography[7, 3]. With this model problem, we will consider four different experiments. In the first, we will consider using the $H^1$ norm for ADMM consensus compared to the $L^2$ norm. In the second, we will demonstrate the effectiveness of using inexact updates to accelerate the global solution. In our third experiment, we will consider the computational cost of the ADMM compared with the monolithic approach on a discrete mesh at multiple refinements with a fixed number of PDE models. Our fourth experiment will analyze the computational cost of the ADMM compared with the monolithic approach with a varying number of PDE models on a mesh of fixed size. Here the monolithic approach references. The term monolithic refers to solving a single large problem without breaking it into smaller sub-problems. The monolithic approach then implemented a traditional INCG descent method, found in Algorithm 3, to directly optimize the cost functional of the inverse problem. With each ADMM inversion, we performed the same inversion using the monolithic approach to compare accuracy, solution time, and computational cost.
To calculate the instance parameter updates and the monolithic approach, we applied the INCG solver found in hIPPYlib, an extensible software framework for large-scale inverse problems governed by PDEs [20]. The update of $z$ in Algorithm 5 was calculated using the PETScTAOSolver built into Fenics[15], a comprehensive library designed for numerical solutions for PDEs.

5.1. Formulating electrical impedance tomography in the continuous setting. Electrical impedance tomography is an imaging modality that relies on inputting an electrical current to a portion of the domain boundary and measuring the resulting electric potential on the rest of the domain’s boundary. The electric potential is dependent on the conductivity of the material throughout the domain. In this example, we consider a compact domain $\Omega \in \mathbb{R}^2$ representing the object to be imaged of and let $\mathcal{M} := H^1(\Omega)$ be the Sobolev space of square-integrable functions with square-integrable gradients. The data fidelity terms $\mathcal{L}_i$ in (2.1) have the form:

$$\mathcal{L}_i(m) = \frac{1}{2} \int_{\Gamma_i} (u_i - d_i)^2 ds,$$

where $\Gamma_i \subset \partial \Omega$ is portion of the boundary where the state variable (electric potential) $u_i$ is measured. The potential $u_i$ solves the electrostatic Maxwell equation:

$$\begin{cases}
-\nabla \cdot e^m \nabla u_i = 0 & x \in \Omega \\
\frac{\partial}{\partial n} u_i = g_i & x \in \Gamma_N^i \\
u_i = 0 & x \in \Gamma_D^i
\end{cases}$$

(5.1)

Here $\sigma := e^m$ is the conductivity of the domain, and $u_i$ is the electric potential resulting from introducing the current $g_i$. $\Gamma_N^i$ denotes the Neumann boundary corresponding to the current injected, and $\Gamma_D^i$ is the Dirichlet boundary corresponding to the electrical ground. $\partial \Omega = \Gamma_N^i \cup \Gamma_D^i$. Suppose then that we perform $q$ measurements with $q$ different currents, resulting in $d_i$.

Our goal is then to find a minimize (5.2) satisfying (5.1).

$$\frac{1}{2q} \sum_{i=1}^{q} \mathcal{L}_i(m) + \mathcal{R}(m),$$

(5.2)

where $\mathcal{R}(m)$ is a combination of Total Variation and $L^2(\Omega)$ regularization defined as

$$\mathcal{R}(m) = \alpha_{TV} \int_{\Omega} |\nabla (m - m_{pr})| \varepsilon dx + \frac{\alpha_{TK}}{2} \int_{\Omega} (m - m_{pr})^2 dx,$$

(5.3)

where $m_{pr} \in \mathcal{M}$ is a reference value for the inversion parameter, $\alpha_{TV}, \alpha_{TK} > 0$ are the regularization parameters. Finally,

$$|\nabla (m - m_{pr})| \varepsilon = \sqrt{(\nabla (m - m_{pr}))^T (\nabla (m - m_{pr})) + \varepsilon}$$

is a smooth approximation to make the TV functional differentiable. The parameter $\varepsilon > 0$ controls the smoothness of the functional.
5.2. Discretization. The unit disc was selected as our domain of interest \( \Omega \). This continuous domain was then discretized with a uniform mesh with triangular elements. On this mesh, we chose our set of basis functions to be continuous piecewise linear finite polynomials for both \( M \) and \( U \). On our coarsest mesh, we then had 8044 degrees of freedom on the parameter and state variables. However, for our experiment in 5.6, we will perform multiple mesh refinements. This will then result in parameter and state variables with 8044, 31816, 71280, and 126428 degrees of freedom.

5.3. Ground truth, synthetic data. The true parameter used was a modified Shepp-Logan Phantom on the unit circle displayed in Figure 5.1 (left).

Next we let the incident current \( g_i \) be given by (5.4)

\[
(5.4) \quad g_i(\theta) = \gamma \exp(-\beta(\theta - \theta_i)^2)
\]

where \( \theta \) is the angle a point on \( \partial \Omega \), \( \theta_i \) dictate the position of the electrical source are dispersed evenly along the boundary, and \( \gamma, \beta > 0 \) are constants dictating the amplitude and decay of the source. We will use \( \gamma = 0.1 \) and \( \beta = 10 \). The Dirichlet boundary \( \Gamma_D \), which acted as electrical ground, was chosen to be a single point on the boundary. The rest of the boundary was considered to be the Neumann boundary, \( \Gamma_N = \partial \Omega \setminus \Gamma_D \).

The true states for models 1, 11, 16 for the \( q = 16 \) case are displayed in Figure 5.1 (right). The electrical source is highlighted with a red sphere, and the electrical ground is highlighted with a blue sphere.

![Figure 5.1](image)

**Fig. 5.1.** True parameter (left) and true states corresponding to sources 1, 11, 16 (right) for EIT problem with \( q = 16 \)

The states were then perturbed by random noise with a standard deviation of \( 0.01 \ast ||u||_\infty \), or one hundredth of the maximum state. To account for this noise we performed reconstructions with regularization decided by \( \alpha_{TV} = 0.1 \), \( \varepsilon = 10^{-4} \) and \( \alpha_{TK} = 0.01 \).

Here we will also use the adaptive \( \rho \) scheme with \( \mu = 2 \) and \( \tau = 3 \). The termination constants are set to \( \varepsilon_a = 10^{-5} \) and \( \varepsilon_r = 2 \cdot 10^{-2} \).

5.4. Reconstruction with \( L^2 \) norm and \( H^1 \) norms. In this section we compare reconstruction performed with the norms given by the \( L^2(\Omega) \) and \( H^1(\Omega) \) norms. \( L^2(\Omega) \) is the space of square integrable functions with a norm given by the square root of this integral. The Sobolev space \( H^1(\Omega) \) is the space of functions such that the norm given in (5.5) is bounded.

\[
(5.5) \quad ||m||_{H^1(\Omega)} = \left( \int_\Omega ||m||^2 + || \nabla m ||^2 dx \right)^{1/2}
\]
For these reconstructions, we fix $q = 16$. The ADMM solver utilized a maximum of 10 global iterations and used an INCG solver with a maximum of 10 iterations to find a more precise estimate of individual parameters at each global iteration. The consensus variable was updated using the PETScTAOSolver implementing a Newton, trust-region method with an absolute tolerance on the gradient of $10^{-12}$, relative gradient tolerance of $10^{-9}$, and at most 10 iterations.

The ADMM solver with the $L^2$ consensus began with $\rho^0 = 1000$ whereas the ADMM solver with the $H^1$ consensus began with $\rho^0 = 0.1$. The higher value of $\rho^0$ for the $L^2$ consensus is needed to give approximately the same starting consensus weight to both solution methods. The solvers had a global tolerance of $10^{-3}$ and a relative tolerance of $10^{-2}$. The $L^2$ consensus solver terminated in 7 iterations and the $H^1$ solver terminated in 10 iterations. The final consensus for the $L^2$ solution is pictured on the left of Figure 5.2 and the final consensus for the $H^1$ solution is on the right.

![Fig. 5.2. Inverted consensus for EIT problem with ADMM solver using $L^2$ and $H^1$ norms](image)

Qualitatively, we can not discern the original image's traits from the $L^2$ reconstruction, which is approximately constant. Compare this to the $H^1$ reconstruction, which is much more faithful to the ground truth. Quantitatively, this $L^2$ reconstruction had a final relative error of .2279 where we define the error in (5.6) and was not reduced across global iterations. The $H^1$ reconstruction had a clear and gradual reduction in error across iterations and had a final relative error of 0.1552. Figure 5.3 displays the error and relative error resulting from the different consensus norms.

\begin{equation}
\text{(5.6) Relative error } = \frac{||m_{true} - m_{MAP}||_{L^2(\Omega)}}{||m_{true}||_{L^2(\Omega)}}
\end{equation}

We can further analyze the effect of the different consensus norms by looking at the primal and dual residuals' behavior for ADMM displayed in Figure 5.4.

Using the $H^1$ norm creates a stable and clear decrease in the residuals compared to the $L^2$ norm. This improvement in the inversion performance and stability demonstrates that the $H^1$ norm is better suited for applying the ADMM to variational problems of this form compared to using the $L^2$ norm. The Table 5.1 summarizes the accuracy of the the $L^2$ and $H^1$ models via the relative error and state misfit. Table 5.1 also outlines the computational cost of each method via the solution time and forward, adjoint, and incremental solves.
5.5. Reconstruction with $H^1$ norm using inexact parameters solutions.

In the previous section, we demonstrated the performance of ADMM by using the $H^1$ norm for consensus reinforcement. We can further improve upon this solution’s performance by solving the inverse problems associated with each PDE model inexactly. This will increase the number of global iteration, but each iteration becomes progressively less expensive.

With this idea, we proceed with an inversion for $q = 16$. The ADMM solver utilized a maximum of 40 global iterations and used an INCG solver with a maximum of 3 iterations for an inexact estimate of the individual parameters at each global iteration. The ADMM solver began with $\rho_0 = 0.1$ and had a global absolute tolerance of $10^{-5}$ and a global relative tolerance of $2 \cdot 10^{-2}$. This solution terminated at 20 iterations when the tolerances were reached and resulted in the final consensus shown in Figure 5.5.

This reconstruction had even better performance than the instance with exact

| $H^1$ | 10 | 6m 50s | 0.1160 | 1.226 | 1946 | 1583 | 33968 |
|-------|----|-------|-------|-------|------|------|-------|
| $L^2$ | 5  | 1m 57s| 0.2249| 3.249 | 1014 | 799  | 4444  |

**Table 5.1**

Comparison between use of $L^2$ and $H^1$ consensus norm in ADMM
solutions for each global iteration parameter. The final relative was 0.1160 and was reduced with each global iteration as seen in Figure 5.6. Similarly, the primal and dual residuals exhibited very nice behaviors with constant reductions, as seen in Figure 5.7. The plots below have both the Inexact and Exact results plotted against the number of forward PDE solves at that stage.

Utilizing inexact solutions for the parameters also lead to a much faster inversion that met convergence requirements. The exact solution took 2 minutes and 59 seconds and terminated. The inexact solution took 2 minutes and 30 seconds and terminated when the convergence requirements were met. The table 5.2 summarizes the performance of using inexact parameter solver over exact parameter solves.
5.6. Comparing the ADMM and monolithic scalability with respect to problem size. We compared the effectiveness of the ADMM with the monolithic approach by performing multiple reconstructions using both approaches on meshes at various levels of refinement. Using a finer mesh meant a larger number of degrees of freedom for both the inversion parameters and state variables. With this, we performed inversions at four different levels of refinement with 8044, 31816, 71280, and 126428 degrees of freedom. We then fixed the number of PDE models to $q = 16$ for these inversions. For both approaches, we also ran their solution in parallel across 8 processes.

For the ADMM method, we utilized a solver with a global absolute tolerance of $10^{-5}$ and a global relative tolerance of $2 \cdot 10^{-2}$. For each global iteration, the local inversion parameter was solved using an INCG solver with 3 iterations, a relative tolerance of $10^{-6}$, and an absolute tolerance of $10^{-9}$. This INCG solver utilized a maximum of 100 conjugate gradient evaluations at each iteration. We also used the $H^1$ norm for consensus reinforcement and began the adaptive $\rho$ scheme with $\rho^0 = 0.1$. The consensus variable was updated using the PETScTAOSolver implementing a Newton, trust-region method with an absolute tolerance on the gradient of $10^{-12}$, relative gradient tolerance of $10^{-9}$, and at most 10 iterations.

We implemented an INCG solver for the monolithic solution with a relative tolerance of $10^{-6}$ and an absolute tolerance of $10^{-2}$. This INCG solver utilized a maximum of 100 conjugate gradient evaluations for each iteration and had a maximum number of 75 iterations.

5.6.1. Reconstructions over varying levels of mesh refinement. Using the ADMM and monolithic approaches, we achieved the following reconstructions at each level of refinement. The top row contains the ADMM inversions corresponding to 8044, 31816, 71280, and 126428 degrees of freedom. The bottom row contains the monolithic inversions corresponding to 8044, 31816, 71280, and 126428 degrees of freedom. The relative error and state misfits are plotted against the number of degrees of freedom in Figure 5.9.
5.6.2. Computational scalability with respect problem size. The time required for each reconstruction is plotted against the number of degrees of freedom in Figure 5.10. The computational cost was assessed in three different ways: tracking how many times the forward problem was solved, how many times the adjoint problem was solved, and how many times the incremental problems were solved. The total number of forward solves, adjoint solves, and incremental solves is plotted against the number of degrees of freedom in Figure 5.11.

The ADMM approach achieved satisfactory accuracy compared to the monolithic approach at varying problem sizes with similar amounts of forward and adjoint solves. However, the ADMM solutions required much fewer incremental evaluations. Having fewer incremental evaluations then reduced the amount of time for the solution time.
Fig. 5.10. Solution time (s) vs number of degrees of freedom

These results are summarized in Table 5.3.

| Degrees of freedom | 8044 | 31816 | 71280 | 126428 |
|--------------------|------|-------|-------|--------|
| Solution time      | 3m 1s/7m 32s | 4m 57s/15m 50s | 14m 12s/17m 47s | 19m 8s/37m 0s |
| Relative error     | 0.1104/0.0864 | 0.1206/0.0954 | 0.1194/0.0952 | 0.1222/0.0985 |
| State misfit       | 1.259/1.249 | 1.277/1.342 | 1.294/1.317 | 1.37/1.314 |
| Forward solves     | 1348/1088 | 903/880 | 1354/976 | 1171/736 |
| Adjoint solves     | 1008/1065 | 672/848 | 1008/864 | 864/704 |
| Incremental solves | 6649/45616 | 3955/31328 | 6351/27712 | 5108/23408 |

5.7. Comparing the ADMM and monolithic scalability with respect to number of forward models. We compared the effectiveness of the ADMM with the monolithic approach by performing multiple reconstructions using both approaches with a varying number of PDE models on a mesh with a fixed number of degrees of freedom. The number of PDE models varied according to \( q = 8, 16, 32, 64 \). The solutions for both approaches were found in parallel across 8 processes. The solver settings were the same as in subsection 5.6.

5.7.1. Reconstructions over with multiple PDE models. Using the ADMM and monolithic approaches, we achieved the following reconstructions at each level of refinement. The top row contains the ADMM inversions corresponding to 8, 16, 32, and 64 PDE models. The bottom row contains the monolithic inversions corresponding to 8, 16, 32, and 64 PDE models. The relative error and state misfits are plotted against the number of PDE models in Figure 5.13.

5.7.2. Computational scalability with respect number of PDE models. The time required for each reconstruction is plotted against the number of PDE models in Figure 5.14. We continue to assess computational cost based on the number of forward, adjoint, and incremental solves. The total number of forward solves, adjoint solves, and incremental solves is plotted against the number of PDE models in Figure 5.15.

The ADMM still achieves satisfactory accuracy in a shorter amount of time for each number of PDE models. Once again, this is is because it requires much fewer
incremental evaluations for each solution. These results are summarized in Table 5.4.

| PDE models | 8     | 16    | 32    | 64    |
|------------|-------|-------|-------|-------|
| Solution time | 52s/4m 13s | 3m 6s/7m 45s | 6m 58s/8m 28s | 15m 44s /16m 26s |
| Relative error | 0.1261/ 0.0981 | 0.1116/0.0817 | 0.1059/0.0817 | 0.1033/0.0707 |
| State misfit | 1.284/1.127 | 1.221/1.242 | 1.264/1.266 | 1.244/1.307 |
| Forward solves | 417/624 | 1414/1072 | 3104/1568 | 6758/3264 |
| Adjoint solves | 312/600 | 1056/1040 | 2304/1504 | 4992/2880 |
| Incremental solves | 1566/25312 | 6792/46360 | 16748/48544 | 40278/88576 |

6. Application to multi-wavelength quantitative photoacoustic tomography problem. This section presents an application of the proposed ADMM method to quantitative photoacoustic tomography (qPACT). qPACT is an emerging medical imaging technique that holds great promise for early cancer diagnosis because it is non-invasive, radiation-free, and inexpensive. qPACT is a hybrid modality that combines endogenous contrast of optical imaging with the high-resolution of ultrasound.
detection technologies to provide maps of total hemoglobin content and oxygen saturation within the tissue[21, 14].

This process requires solving a series of two inverse problems and thus involves two separate stages. The process begins with a fast laser pulse in the infrared range being sent into the object of interest. The underlying material then absorbs this optical energy and generates heat and a local increase pressure distribution. This pressure distribution then transitions into acoustic waves that can then be observed at the boundary of the domain. This process can be viewed as two inverse problems. The first involves reconstructing for the initial pressure distribution given measurements on the boundary of the domain. The second involves reconstructing the optical properties of the tissue based on the initial pressure distribution. Here we only worry about the second inverse problem of reconstructing tissue composition given the initial pressure distribution.
6.1. Formulation of the qPACT problem. Let $p_0$ denote this initial pressure distribution. $p_0$ is related to the absorption coefficient $\mu_a$ of the domain via (6.1).

\begin{equation}
\frac{d}{\Gamma} = \mu_a \phi
\end{equation}

$\Gamma$ is the Grunesian parameter, which we will treat as a known value of constant 1. The fluence $\phi$ is then determined as the solution of the PDE relationship below, known as the diffusion approximation.

\begin{equation}
\begin{aligned}
-\nabla \cdot \left[ \frac{1}{3(\mu_a + \mu'_s)} \nabla \phi + \mu_a \phi \right] &= 0 & \quad x \in \Omega \\
\frac{1}{3(\mu_a + \mu'_s)} \frac{\partial \phi}{\partial n} + \frac{1}{2} \phi &= \frac{1}{2} \phi_0 & \quad x \in \partial \Omega
\end{aligned}
\end{equation}

where $\mu'_s$ is the reduced scattering coefficient and $\phi_0$ is the intensity of the incident illumination. Here $\mu'_s = (1 - g)\mu_s$, where $\mu_s$ is the scattering coefficient and $g$ is the anisotropy factor. We will fix $g = .9$. The absorption coefficient $\mu_a$ is a linear combination of basis materials called chromophores and reads

\begin{equation}
\mu_a = \sum_i \epsilon_i(\lambda) c_i
\end{equation}

$\epsilon_i$ is a known function of the incident wavelength and $c_i$ is the concentration of the $i$-th chromophore.

We will be attempting to reconstruct for a 2d maximum intensity projection of a human breast. Using human breast tissues as our domain means that we can safely limit our chromophores to deoxygenated hemoglobin $c_{hb}$ and oxygenated hemoglobin $c_{hb02}$. These two values are then related to the oxygen saturation $s$ and the total hemoglobin concentration $c_{thb}$ via (6.4).

\begin{equation}
\begin{aligned}
c_{hb} &= (1 - s) \cdot c_{thb} \\
c_{hb02} &= s \cdot c_{thb}
\end{aligned}
\end{equation}

We can then invert for $s$, $c_{thb}$, and $\mu_s$ with a PDE model associated with each incident wavelength. For each PDE model, there will be an associated fluence $\phi_i$, \ldots
which acts as the state variable. Letting \( d_i \) be the observations associated with the \( i \)-th incident wavelength, we can then form the data fidelity term

\[
\frac{1}{q} \sum_{i=1}^{d} C_i(s, c_{thb}, \mu'_s) = \frac{1}{q} \sum_{i=1}^{d} || \ln(\mu_{s,i} \phi_i) - \ln(d_i) ||^2
\]

6.2. Inversion results. We then perform an inversion with measurements from 757, 800, and 850 wavelengths with uniform intensity around the 2d boundary. The ground truth values for \( s, c_{thb}, c_{hb}, \) and \( c_{thbo} \) are shown in the top row of Figure 6.2. We ran forward model for each wavelength and perturbed each of the data measurements by white Gaussian noise with a standard deviation equal to a hundredth of their maximum value. We then used the regularization shown in (6.5).

\[
R(s, c_{thb}, \mu'_s) = \gamma_s \int_{\Omega} || \nabla s ||^2 dx + \delta_s \int_{\Omega} || s ||^2 dx
\]

\[
\gamma_{c_{thb}} \int_{\Omega} || \nabla c_{thb} ||^2 dx + \delta_{c_{thb}} \int_{\Omega} || c_{thb} ||^2 dx
\]

\[
\gamma_{\mu'_s} \int_{\Omega} || \nabla \mu'_s ||^2 dx + \delta_{\mu'_s} \int_{\Omega} || \mu'_s ||^2 dx
\]

Using ADMM is of much interest because it can iteratively solve stable subproblems to continuously advance to the true solution while appropriately handling the complex regularization function. We then applied the ADMM to solve this problem with...
the three different wavelength and a regularization determined by $\gamma_s = 0.05, \delta_s = 0.001, \gamma_{c_{thb}} = 0.005, \delta_{c_{thb}} = 10^{-6}, \gamma_{c_{hb}} = 10, \delta_{c_{hb}} = 10$, and $\varepsilon = 10^{-6}$. In doing this, we set an absolute global tolerance of $10^{-4}$ and a relative global tolerance of $10^{-3}$. The optimal values of $m^k$ were calculated using an INCG solver with at most 50 iterations, a relative tolerance of $1e^{-6}$, and an absolute tolerance of $10^{-3}$. The number of iterations on this solver was more than needed and always converged within 30 iterations. The consensus variable was updated using a PETScTAOSolver solver with a relative tolerance of $1e^{-9}$ and an absolute tolerance of $1e^{-12}$. We implemented the adaptive $\rho$ scheme with $\mu = 4$ and $\tau = 2$. We were then able to reconstruct for the parameters shown in the bottom row of Figure 6.2.

This reconstruction was very accurate and had minimal errors. The global relative error on the saturation was only 0.061, and the global relative error on the total hemoglobin concentration was only 0.097. These errors were even lower on the arteries and veins, these places with high hemoglobin content and saturation of 0.95 for arteries and 0.7 for veins. The percentage errors on the arteries were 0.040 for the saturation and 0.061 for the total hemoglobin content. On the veins, the percentage errors were 0.050 and 0.055. These low errors mean that we can accurately identify arteries and veins using this reconstruction process. The error on this reconstruction is summarized in Table 6.1.

| Region | $L^2$ norm error | relative error |
|--------|------------------|---------------|
| Global | 5.041813126/4.669520028 | 0.060942239/0.096811564 |
| Artery | 1.207798494/1.685043185 | 0.03977134/0.06131227 |
| Vein  | 1.543544399/2.028329233 | 0.050015734/0.054519924 |
| Inner | 2.364520795/2.940342136 | 0.050783492/0.07463648 |
| Outer | 4.452967663/3.627506777 | 0.065116128/0.130352342 |

This reconstruction saw a continuous decrease in the residuals with each iteration.
These residuals are shown in Figure 6.3, and the $\rho$ for each iteration is shown in Figure 6.4.

Fig. 6.3. Primal and dual residuals from the qPACT reconstruction

Fig. 6.4. $\rho$ values of each iteration of qPACT reconstruction

7. Conclusions. In this paper, we presented a framework for solving inverse problems governed by PDE forward models using ADMM. Through our numerical studies with electrical impedance tomography, we have shown the natural way in which ADMM can solve problems involving several large-scale PDE models with nonsmooth regularization. This ADMM solution method significantly reduced these solutions’ computational costs while still achieving satisfactory accuracy at various scales. This framework preserves consistency with the infinite formulation of these inverse problems and utilizes the underlying Hilbert spaces’ norm to enforce the consensus condition. The effectiveness of the ADMM framework was also demonstrated on a complex multiphysics problem related to photoacoustic tomography. Solving the photoacoustic tomography problem accurately and efficiently shows the viability of the ADMM framework outside the model problem related to electrical impedance tomography. In the future, we plan to improve upon this framework in two ways. First, we plan on implementing more advanced solvers for the total variation denoising problem, including the primal-dual method in [6] and the proximal splitting methods in [13]. Second, we plan on implementing the ADMM process on several processors, with
each PDE model being handled by its own set of processors. Splitting PDE models along processor sets would allow every parameter associated with a PDE model to be updated simultaneously and accelerate the entire process.

REFERENCES

[1] M. Afonso, J. Bioucas-Dias, and Figueiredo M., An Augmented Lagrangian Approach to the Constrained Optimization Formulation of Imaging Inverse Problems, arXiv e-prints, (2011).

[2] G. Biros and O. Ghattas, Inexactness issues in the Lagrange-Newton-Krylov-Schur method for PDE-constrained optimization, in Large-Scale PDE-Constrained Optimization, L. T. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, eds., vol. 30 of Lecture Notes in Computational Science and Engineering, Springer, 2003, pp. 93–114.

[3] L. Borcea, G. Gray, and Y. Zhang, Variationally constrained numerical solution of electrical impedance tomography, Tech. Report TR02-13, Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005–1892, 2003. available electronically from http://www.caam.rice.edu/caam/trs/trtitles.html.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, Foundation and Trends in Machine Learning, Vol. 3, No. 1 (2010).

[5] 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 (2013), pp. A2494–A2523.

[6] Tony F. Chan, Gene H. Golub, and Pep Mulet, A nonlinear primal-dual method for total variation-based image restoration, SIAM Journal on Scientific Computing, 20 (1999), pp. 1964–1977.

[7] M. Cheney, D. Isaacson, and J. C. Newell, Electrical impedance tomography, SIAM Review, 41 (1999), pp. 85–101.

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

[9] Stanley C. Eisenstat and Homer F. Walker, Choosing the forcing terms in an inexact Newton method, SIAM Journal on Scientific Computing, 17 (1996), pp. 16–32.

[10] Alexandre Ern and Jean-Luc Guermond, Theory and Practice of Finite Elements, vol. 159 of Applied Mathematical Sciences, Springer-Verlag, 2004.

[11]点亮 Wu Fung and Lars Ruthotto, An uncertainty-weighted asynchronous admm method for parallel pde parameter estimation, SIAM Journal on Scientific Computing, 41 (2019), pp. S129–S148.

[12] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximations, Computers and Mathematics with Applications, Vol. 2, No. 1 (1976).

[13] Marc Herrmann, Roland Herzog, Stephan Schmidt, José Vidal-Núñez, and Gerd Wachsmuth, Discrete total variation with finite elements and applications to imaging, Journal of Mathematical Imaging and Vision, 61 (2019), pp. 411–431.

[14] A. Klose and E. Larsen, Light transport in biological tissue based on the simplified spherical harmonics equations, Journal of Computational Physics, Vol. 220, No. 1 (2006).

[15] Anders Logg, Kent-Andre Mardal, and Garth N. Wells, eds., Automated Solution of Differential Equations by the Finite Element Method, vol. 84 of Lecture Notes in Computational Science and Engineering, Springer, 2012.

[16] J. Tinsley Oden Oden and Junuthula N. Reddy, An introduction to the Mathematical theory of finite elements, Wiley Interscience, New York, 1976.

[17] R. Rockafellar, Augmented Lagrangians and Applications of the Proximal Point Algorithm in Convex Programming, Mathematics of Operations Research, (1976).

[18] Leonid I. Rudin, Stanley Osher, and Emad Fatemi, Nonlinear total variation based noise removal algorithms, Physica D. Nonlinear Phenomena, 60 (1992), pp. 259–268.

[19] G. Strang and G. J. Fix, An Analysis of the Finite Element Method, Wellesley-Cambridge Press, Wellesley, MA, 1988.

[20] U. Villa, N. Petra, and O. Ghattas, hIPPYlib: An Extensible Software Framework for Large-Scale Inverse Problems Governed by PDEs; Part I: Deterministic Inversion and Linearized Bayesian Inference, ACM Transactions on Mathematical Software, in print (2020).

[21] K. Wang and M. Anastasio, Photoacoustic and thermoacoustic tomography: image formation
principles, Handbook of Mathematical Methods in Imaging, (2015).

[22] Y. Wang, J. Yang, W. Yin, and Y. Zhang, A New Alternating Minimization Algorithm for Total Variation Image Reconstruction, SIAM Journal on Imaging Sciences, (2007).

[23] Y. Wang, W. Yin, and J. Zeng, Global Convergence of ADMM in Nonconvex Nonsmooth Optimization, Journal of Scientific Computing, (2018).

[24] B. Wohlberg, Challenges and Opportunities in Using Automatic Differentiation with Object-Oriented Toolkits for Scientific Computing, arXiv e-prints, (2017).

[25] Toktam Zand, Hamid R Shahkooei, Alison Malcolm, Ali Gholami, and Alan Richardson, Consensus optimization of total variation–based reverse time migration, Computational Geosciences, (2020).