

Inexact Interior-Point Method for Pde-Constrained Nonlinear Optimization

Marcus J. Grote, Johannes Huber,
Drosos Kourounis, Olaf Schenk

Institute of Mathematics
University of Basel
Rheinsprung 21
CH - 4051 Basel
Switzerland

Preprint No. 2013-15
May, 2013

www.math.unibas.ch

INEXACT INTERIOR-POINT METHOD FOR PDE-CONSTRAINED NONLINEAR OPTIMIZATION*

MARCUS J. GROTE[†], JOHANNES HUBER[‡], DROSOS KOUROUNIS[§], AND
OLAF SCHENK[¶]

Abstract. Starting from the inexact interior-point framework from Curtis et al. [*Mathematical Programming Series B*, 32(6), pp. 3447-3475, 2012], we propose an effective reduced-space preconditioner for the full Lagrangian Hessian matrix needed at each Newton iteration. Together they yield a scalable, robust and highly parallel method for the numerical solution of large-scale nonconvex PDE-constrained optimization problems with inequality constraints. Because it uses the full Hessian matrix, modifying it whenever needed, the method is not only globally convergent, but also converges fast locally. Our preconditioner is not tailored to any particular class of PDEs or constraints, but instead judiciously exploits the sparsity structure of the Hessian. Numerical examples from PDE-constrained optimal control, parameter estimation and full waveform inversion demonstrate the robustness and efficiency of the method, even in the presence of active inequality constraints.

Key words. PDE-constrained optimization, nonconvex programming, inequality constraints, inexact interior-point method, preconditioning, KKT system, optimal control, seismic imaging, full-waveform inversion

AMS subject classifications. 49K20, 49J20, 65J22, 90C06, 90C51

DOI.

1. Introduction. Throughout science and engineering, mathematical models of physical systems are extensively expressed as partial differential equations (PDEs). As computational methods for the solution of PDEs reach a mature stage, much interest in current research is shifting toward optimal design, optimal control, and parameter estimation of systems governed by PDEs. Despite their seeming disparity, all these problems fit the class of *PDE-constrained optimization*, where (unknown) PDE parameters, u – initial or boundary data, source terms, material coefficients, or domain geometry – must be determined by minimizing an appropriate objective functional

$$(1.1) \quad \begin{aligned} \min_{y,u} \mathcal{F}(y, u) \\ \text{s.t. } \mathcal{A}_k(y_k, u) = 0, \quad \text{for } k = 1, \dots, N_E \\ u_- \leq u \leq u_+. \end{aligned}$$

Here $y = (y_1, y_2, \dots, y_{N_E})$ denotes a collection of state variables y_k that satisfy (possibly nonlinear) PDEs, each involving u . In *optimal control*, for instance, u appears as a source term in every PDE; hence when the differential operators \mathcal{A}_k are linear in y_k , y will also depend linearly on u . In *parameter estimation*, however, the partial

*Received by the editors May 15, 2013; accepted for publication (in revised form) July xx, 20xx; published electronically November 26, 20xx. [sisc/31-2/xxxx.html](#)

[†]Department of Mathematics and Computer Science, University of Basel, Rheinsprung 21, CH-4051 Basel, Switzerland (marcus.grote@unibas.ch).

[‡]Department of Mathematics and Computer Science, University of Basel, Rheinsprung 21, CH-4051 Basel, Switzerland (johannes.huber@unibas.ch).

[§]Institute of Computational Science, Università della Svizzera italiana, CH-6900 Lugano, Switzerland (drosos.kourounis@usi.ch).

[¶]Institute of Computational Science, Università della Svizzera italiana, CH-6900 Lugano, Switzerland (olaf.schenk@usi.ch).

differential operator \mathcal{A}_k itself also depends on the model parameters, u ; hence even when \mathcal{A}_k is linear in y_k , y_k usually depends nonlinearly on u yielding a nonconvex optimization problem. As a consequence, this *inverse problem* is often significantly more difficult to solve than the *forward problem* that consists of the mere computation of y for a given u . It is often ill-posed, requires regularization, and yet can still have multiple local solutions. The inequality constraints in (1.1) not only permit the imposition of physical bounds on u , but also the inclusion of information available a priori. Thus, inequality bounds exclude unphysical solutions and prevent algorithms from becoming trapped too easily in a (false) local minimum.

Numerical methods for PDE-constrained nonlinear optimization with inequality constraints usually fall into one of two categories: active-set strategies and *interior-point (IP)* methods. Because of the combinatorial complexity of active-set algorithms, we opt for interior-point methods instead [7, 18, 17], which we apply to an appropriate finite-dimensional discrete version of (1.1).

Since the PDE constraints in (1.1) are only coupled through u , it may seem at first attractive to eliminate the state variables y_1, \dots, y_{N_E} via the PDE constraints. Known as *reduced-space* optimization, that approach capitalizes on well-established methods and software for solving the (nonlinear) PDEs [15, 16, 20, 27]. However, the resulting Hessian matrices needed by Newton's method are dense and cannot be formed explicitly. It requires the repeated exact solution of the state (and adjoint) equations, even during initial steps when the control variables are still far from their optimum, which is computationally inefficient. Moreover, the evaluation of the objective function during the line-search phase becomes as expensive as the solution of the full forward problem. Given those difficulties it is not surprising that most numerical methods for large-scale PDE-constrained optimization are based on gradients alone, thereby avoiding Hessian matrices entirely. Unlike Newton methods, however, gradient-based methods, such as nonlinear conjugate gradients or limited memory quasi-Newton methods [24], do not scale well with increasing problem size, suffering a reduction from quadratic to linear asymptotic convergence, and often fail to converge for ill-conditioned problems.

In contrast, *full-space* optimization [4, 5, 13, 25] treats at once both state and control as independent optimization variables that are coupled through equality or inequality constraints. Then the Hessian matrix, though very large, is also very sparse and therefore ideally suited for iterative solvers. For nonconvex problems, however, the projection of the Hessian onto the nullspace of the constraint Jacobian may not be positive definite; hence, the resulting search direction may point towards a saddle-point or a maximizer, instead of a minimizer. To enforce a direction of descent, the Hessian matrix must be appropriately modified. When a matrix factorization is explicitly available, the number of positive, negative and zero eigenvalues is easily obtained and can be used to modify the Hessian, if needed, and thus enforce a descent direction. Iterative methods, however, do not provide such information. As a consequence, most work on inexact Newton methods for large-scale PDE-constrained optimization usually avoids the indefiniteness of the Hessian either by neglecting part of it (Gauss-Newton approximation) and thus sacrificing speed of convergence [3], or simply by assuming convexity of the objective function [2].

Recently, Curtis et al. [9, 11] proposed a full-space inexact interior-point method that guarantees global convergence, while retaining the full Lagrangian Hessian information and thus the fast local convergence of Newton's method. The method was extended to inequality constraints in [12] and further enhanced for additional

efficiency in [10]. There, an algebraic multilevel preconditioner for general sparse indefinite linear systems was used within the inexact Newton step [6], and it led to significant speed-up over an algorithm based on direct factorization of the primal-dual system. Though encouraging, that approach does not extend to truly large-scale problems with multiple PDE constraints, because the preconditioner requires assembly and global reordering of the entire Hessian matrix; hence, it is actually restricted to shared-memory architectures.

Following [10], we present in Section 2 the inexact interior-point (IIP) framework and recall the salient points of the algorithm. Next in Section 3, we describe our reduced-space preconditioner (RSP) for the full Hessian matrix. Together they yield the RSP-IIP method, which we apply to a series of numerical examples in Section 4 to demonstrate its efficiency and robustness. Here we consider both PDE-constrained control and inverse problems, either in two or three space dimensions: all but one are nonconvex. Finally, we summarize our findings in Section 5.

2. Interior-Point Framework. Numerical approximation of (1.1) leads to its discrete counterpart,

$$(2.1) \quad \begin{aligned} \min_{\mathbf{y}, \mathbf{u}} F(\mathbf{y}, \mathbf{u}) &= \frac{1}{2} \sum_{k=1}^{N_E} \|\mathbf{V} \mathbf{y}_k - \hat{\mathbf{y}}_k\|^2 + \frac{\alpha}{2} \mathbf{R}(\mathbf{u}) \\ \text{s.t. } \mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) &= 0, \quad \text{for } k = 1, \dots, N_E \\ \mathbf{u}_- &\leq \mathbf{u} \leq \mathbf{u}_+, \end{aligned}$$

where bold symbols denote finite-dimensional vector or matrix approximations of the corresponding functions or operators from the continuous formulation. The objective function now consists of two parts. The misfit term measures the deviation from some desired state or measurements $\hat{\mathbf{y}}_k$, where \mathbf{V} is a projection or evaluation matrix. The Tikhonov regularization term $\mathbf{R}(\mathbf{u})$ with parameter α is standard and removes ill-posedness.

Following a classical interior-point (IP) strategy, we now introduce slack variables $\mathbf{s} \in \mathbb{R}^{2n_u}$, $\mathbf{s} \geq \mathbf{0}$ and a barrier term with parameter, $\mu > 0$, for the inequality constraints. Then (2.1) is solved through a sequence of barrier subproblems

$$(2.2) \quad \begin{aligned} \min_{\mathbf{y}, \mathbf{u}, \mathbf{s}} \varphi(\mathbf{y}, \mathbf{u}, \mathbf{s}; \mu) &:= \frac{1}{2} \sum_{k=1}^{N_E} \|\mathbf{V} \mathbf{y}_k - \hat{\mathbf{y}}_k\|^2 + \frac{\alpha}{2} \mathbf{R}(\mathbf{u}) - \mu \sum_{i=1}^{2n_u} \log(s_i) \\ \text{s.t. } \mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) &= 0, \quad \text{for } k = 1, \dots, N_E, \\ \mathbf{s} &= \begin{pmatrix} \mathbf{u} - \mathbf{u}_- \\ \mathbf{u}_+ - \mathbf{u} \end{pmatrix} \end{aligned}$$

for a sequence of barrier parameters $\mu \downarrow 0$. If the objective function and the constraints are sufficiently smooth, the limit of the corresponding solutions of (2.2) satisfies first-order optimality conditions for (2.1) when the constraint Jacobian has full rank [24].

Solutions of (2.2) are critical points of the Lagrangian function

$$\begin{aligned} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) &= \frac{1}{2} \sum_{k=1}^{N_E} \|\mathbf{V} \mathbf{y}_k - \hat{\mathbf{y}}_k\|^2 + \frac{\alpha}{2} \mathbf{R}(\mathbf{u}) - \mu \sum_{i=1}^{2n_u} \log(s_i) \\ &\quad + \sum_{k=1}^{N_E} \boldsymbol{\lambda}_k^\top \mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) + \boldsymbol{\lambda}_I^\top \left(\begin{pmatrix} \mathbf{u} - \mathbf{u}_- \\ \mathbf{u}_+ - \mathbf{u} \end{pmatrix} - \mathbf{s} \right), \end{aligned}$$

and thus satisfy the Karush-Kuhn-Tucker (KKT) conditions

$$\begin{aligned}
(2.3) \quad & \nabla_{\mathbf{y}_k} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) = \mathbf{V}^\top (\mathbf{V} \mathbf{y}_k - \hat{\mathbf{y}}_k) + \mathbf{J}_{\mathbf{y}_k}^\top \boldsymbol{\lambda}_k = \mathbf{0} \quad \text{for } k = 1, \dots, N_E, \\
& \nabla_{\mathbf{u}} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) = \frac{\alpha}{2} \nabla_{\mathbf{u}} \mathbf{R} + \sum_{k=1}^{N_E} \mathbf{J}_{k\mathbf{u}}^\top \boldsymbol{\lambda}_k + \hat{\mathbf{I}}^\top \boldsymbol{\lambda}_{\mathcal{I}} = \mathbf{0}, \\
& \mathbf{S} \nabla_{\mathbf{s}} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) = -\mu \mathbf{e} - \mathbf{S} \boldsymbol{\lambda}_{\mathcal{I}} = \mathbf{0}, \\
& \nabla_{\boldsymbol{\lambda}_k} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) = \mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) = \mathbf{0} \quad \text{for } k = 1, \dots, N_E, \\
& \nabla_{\boldsymbol{\lambda}_{\mathcal{I}}} \mathbf{L}(\mathbf{y}, \mathbf{u}, \mathbf{s}, \boldsymbol{\lambda}) = \begin{pmatrix} \mathbf{u} - \mathbf{u}_- \\ \mathbf{u}_+ - \mathbf{u} \end{pmatrix} - \mathbf{s} = \mathbf{0},
\end{aligned}$$

where pre-multiplication by $\mathbf{S} = \text{diag}(\mathbf{s})$ in the third equation leads to a primal-dual IP method [24]. Moreover, $\mathbf{J}_{\mathbf{y}_k} = \nabla_{\mathbf{y}_k} \mathbf{A}_k(\mathbf{y}_k, \mathbf{u})$ and $\mathbf{J}_{k\mathbf{u}} = \nabla_{\mathbf{u}} \mathbf{A}_k(\mathbf{y}_k, \mathbf{u})$ denote the Jacobians of the k -th PDE constraint with respect to \mathbf{y}_k and \mathbf{u} , respectively, $\hat{\mathbf{I}}^\top = [\mathbf{I} \quad -\mathbf{I}]^\top$ denotes the Jacobians of the inequality constraints, and \mathbf{e} is a vector with all entries equal to one. The interior-point method terminates, once the current iterate (\mathbf{y}, \mathbf{u}) satisfies (2.3) with $\mu = 0$ within a prescribed tolerance ε_{tol} , that is

$$(2.4) \quad \|\nabla \mathbf{L}\|_\infty \leq \varepsilon_{tol}$$

with $\mu = 0$.

At an iterate (\mathbf{y}, \mathbf{u}) , we compute a primal-dual search direction $(\mathbf{d}_p, \mathbf{d}_d)$, $\mathbf{d}_p = (\mathbf{d}_y, \mathbf{d}_u, \mathbf{d}_s)$ and $\mathbf{d}_d = (\mathbf{d}_{\boldsymbol{\lambda}_\varepsilon}, \mathbf{d}_{\boldsymbol{\lambda}_{\mathcal{I}}})$, which satisfies appropriate conditions for guaranteeing global convergence [10]. The computation of the search direction is based on Newton's method applied to the KKT conditions (2.3) and hence involves the full Lagrangian Hessian, \mathbf{H} . To simplify notation, we now group the primal and dual variables as $\mathbf{z} = (\mathbf{y}, \mathbf{u}, \mathbf{s})$ and $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_\varepsilon, \boldsymbol{\lambda}_{\mathcal{I}})$, where $\boldsymbol{\lambda}_\varepsilon$ denotes the vector of Lagrange multipliers $\boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_{N_E}$ associated with the PDE (equality) constraints. We further let

$$\nabla_p \mathbf{L} = \begin{pmatrix} \nabla_{\mathbf{y}} \mathbf{L} \\ \nabla_{\mathbf{u}} \mathbf{L} \\ \boldsymbol{\Sigma} \nabla_{\mathbf{s}} \mathbf{L} \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} \nabla_{\boldsymbol{\lambda}_\varepsilon} \mathbf{L} \\ \nabla_{\boldsymbol{\lambda}_{\mathcal{I}}} \mathbf{L} \end{pmatrix}, \quad \mathbf{J} = \begin{pmatrix} \mathbf{J}_y & \mathbf{J}_u & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{I}} & -\boldsymbol{\Sigma} \end{pmatrix},$$

and $\boldsymbol{\Sigma}$ denote a positive definite diagonal scaling matrix for the slack variables.

For nonconvex problems, the projection of \mathbf{H} onto the nullspace of the constraint Jacobian may not be positive definite; hence, the search direction $(\mathbf{d}_p, \mathbf{d}_d)$ may point towards a saddle-point or a maximizer, instead of a minimizer. To enforce a direction of descent, we regularize \mathbf{H} and modify the first three diagonal blocks of the Hessian by adding a multiple of the identity matrix $\delta \mathbf{I}$. Thus at each Newton step we actually solve the linear system

$$(2.5) \quad \begin{pmatrix} \mathbf{H}_{yy} + \delta \mathbf{I} & \mathbf{H}_{yu} & \mathbf{0} & \mathbf{J}_y^\top & \mathbf{0} \\ \mathbf{H}_{yu}^\top & \mathbf{H}_{uu} + \delta \mathbf{I} & \mathbf{0} & \mathbf{J}_u^\top & \hat{\mathbf{I}}^\top \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}(\mathbf{S}^{-1} \boldsymbol{\Lambda}_{\mathcal{I}} + \delta \mathbf{I}) \boldsymbol{\Sigma} & \mathbf{0} & -\boldsymbol{\Sigma} \\ \mathbf{J}_y & \mathbf{J}_u & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{I}} & -\boldsymbol{\Sigma} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{d}_y \\ \mathbf{d}_u \\ \mathbf{d}_s \\ \mathbf{d}_{\boldsymbol{\lambda}_\varepsilon} \\ \mathbf{d}_{\boldsymbol{\lambda}_{\mathcal{I}}} \end{pmatrix} = - \begin{pmatrix} \nabla_{\mathbf{y}_k} \mathbf{L} \\ \nabla_{\mathbf{u}} \mathbf{L} \\ \boldsymbol{\Sigma} \nabla_{\mathbf{s}} \mathbf{L} \\ \nabla_{\boldsymbol{\lambda}_\varepsilon} \mathbf{L} \\ \nabla_{\boldsymbol{\lambda}_{\mathcal{I}}} \mathbf{L} \end{pmatrix},$$

where $\boldsymbol{\Lambda}_{\mathcal{I}} = \text{diag}(\boldsymbol{\lambda}_{\mathcal{I}})$ are the Lagrange multipliers associated with the inequality constraints. The linear system (2.5) is very large but also very sparse and thus well-suited for iterative solvers. Clearly an effective preconditioner is essential to keep the

number of iterations low. In Section 3, we shall propose a general purpose effective and parallel preconditioner.

2.1. Termination criteria. When we replace the direct solution of (2.5) by an iterative method, both error control and the handling of nonconvexity become non-trivial. Indeed if (2.5) is no longer solved exactly, how large an error will nonetheless guarantee global convergence? Moreover, when a matrix factorization is explicitly available, the number of positive, negative and zero eigenvalues is easily obtained and can be used to modify the Hessian, if needed, and thus enforce a descent direction. Iterative methods, however, do not provide such information. Answers to these two questions were given in [11, 9, 12], where a series of sufficient merit function approximation reduction termination tests (SMART tests, for short) were devised that allow inexact solutions of (2.5) and yet guarantee global convergence.

Of central importance in those termination tests are the residual vectors

$$\begin{pmatrix} \mathbf{r}_d \\ \mathbf{r}_p \end{pmatrix} = \begin{pmatrix} \mathbf{H}_{pp} & \mathbf{J}^\top \\ \mathbf{J} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{d}_p \\ \mathbf{d}_d \end{pmatrix} + \begin{pmatrix} \nabla_p \mathbf{L} \\ \mathbf{c} \end{pmatrix},$$

where

$$\mathbf{H}_{pp} = \begin{pmatrix} \mathbf{H}_{yy} + \delta \mathbf{I} & \mathbf{H}_{yu} & \mathbf{0} \\ \mathbf{H}_{yu}^\top & \mathbf{H}_{uu} + \delta \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Sigma(\mathbf{S}^{-1} \Lambda_{\mathcal{I}} + \delta \mathbf{I}) \Sigma \end{pmatrix}$$

as well as the primal-dual relative residuals of (2.5),

$$\Psi := \left\| \begin{pmatrix} \mathbf{r}_d \\ \mathbf{r}_p \end{pmatrix} \right\| / \left\| \begin{pmatrix} \nabla_p \mathbf{L} \\ \mathbf{c} \end{pmatrix} \right\|.$$

For convex problems the algorithm can focus exclusively on Ψ , terminating the calculation of $(\mathbf{d}_p, \mathbf{d}_d)$ whenever this value is below a threshold. For nonconvex problems, however, the priority is to find solutions to (2.3) that correspond to minimizers, not saddle-points or maximizers. The methods developed in [11, 9, 12] therefore include additional conditions and procedures that enforce convergence toward minimizers of (2.1). These additional conditions involve a local model of the merit function ϕ , denoted as

$$m(\mathbf{d}_p; \mu, \pi) := \phi(\mathbf{z}; \mu, \pi) + \nabla_{\mathbf{z}} \varphi(\mathbf{z})^\top \mathbf{d}_p + \pi \|\mathbf{c} + \mathbf{J} \mathbf{d}_p\|$$

and the reduction in this model yielded by \mathbf{d}_p , which is defined as

$$(2.6) \quad \begin{aligned} \Delta m(\mathbf{d}_p; \mu, \pi) &:= m(\mathbf{0}; \mu, \pi) - m(\mathbf{d}_p; \mu, \pi) \\ &= -\nabla_{\mathbf{z}} \varphi(\mathbf{z})^\top \mathbf{d}_p + \pi (\|\mathbf{c}\| - \|\mathbf{c} + \mathbf{J} \mathbf{d}_p\|). \end{aligned}$$

Once the primal-dual search direction $(\mathbf{d}_p, \mathbf{d}_d)$ has been determined, we compute the scaled direction $\tilde{\mathbf{d}}_p = (\mathbf{d}_y, \mathbf{d}_u, \Sigma \mathbf{d}_s)$ along which a line search is performed. The line search involves two conditions. First, to maintain positivity of the slacks, a step size $\alpha^{\max} \in (0, 1]$ satisfying

$$(2.7) \quad \mathbf{s} + \alpha^{\max} \Sigma \mathbf{d}_s \geq (1 - \eta_1) \mathbf{s}$$

is determined for a constant $\eta_1 \in (0, 1)$; we use $\eta_1 = \max(0.99, 1 - \mu)$, which converges to one as $\mu \downarrow 0$ and thus preserves fast local convergence. Second, the algorithm

Algorithm 1 Interior-Point Framework

-
- 1: (Initialization) Choose line search parameters $\eta_1, \eta_2 \in (0, 1)$, $\varepsilon_{tol} > 0$, an initial barrier parameter $\mu > 0$, and an initial penalty parameter $\pi > 0$. Initialize $(\mathbf{z}, \boldsymbol{\lambda})$ so that the slack variables satisfy $\mathbf{s} > \mathbf{0}$ and $\mathbf{s} \geq \begin{pmatrix} \mathbf{u} - \mathbf{u}_- \\ \mathbf{u}_+ - \mathbf{u} \end{pmatrix}$.
 - 2: (Tests for convergence) If (2.4) is satisfied with $\mu = 0$ for tolerance ε_{tol} , then terminate and return \mathbf{y} and \mathbf{u} as an optimal solution. Else if (\mathbf{y}, \mathbf{u}) is an infeasible stationary point with $\mathbf{c} \neq \mathbf{0}$, then terminate with a corresponding notification.
 - 3: (Barrier parameter update) If (2.4) with ε_{tol} replaced by $\kappa_\varepsilon \mu$ (we choose $\kappa_\varepsilon = 10$) is satisfied, that is convergence for (2.2) is reached, then decrease the barrier parameter $\mu > 0$, reset $\pi > 0$, and go to step 2.
 - 4: (Search direction computation) Compute $(\mathbf{d}_p, \mathbf{d}_d)$ from (2.5) and update π according to (Algorithm 2 or Algorithm 3, [10]). Set the search direction to $\tilde{\mathbf{d}}_p := (\mathbf{d}_y, \mathbf{d}_u, \boldsymbol{\Sigma} \mathbf{d}_s)$.
 - 5: (Line search) If $\mathbf{d}_p = \mathbf{0}$, then $\alpha := 1$. Else, let α^{\max} be the largest value in $(0, 1]$ satisfying (2.7) and let l be the smallest value in \mathbb{N}_0 such that $\alpha := 2^{-l} \alpha^{\max}$ satisfies (2.8).
 - 6: (Iterate update) Set $\mathbf{z} := \mathbf{z} + \alpha \tilde{\mathbf{d}}_p$, $\mathbf{s} := \max\{\mathbf{s}, \begin{pmatrix} \mathbf{u} - \mathbf{u}_- \\ \mathbf{u}_+ - \mathbf{u} \end{pmatrix}\}$, $\boldsymbol{\lambda} := \boldsymbol{\lambda} + \beta \mathbf{d}_d$ with β according to (2.9), and go to step 3.
-

backtracks from this value to compute the step length $\alpha \in (0, \alpha^{\max}]$ yielding sufficient decrease in the merit function

$$(2.8) \quad \phi(\mathbf{z}; \mu, \pi) := \varphi(\mathbf{z}; \mu) + \pi \|\mathbf{c}\|,$$

where the merit parameter, $\pi > 0$, is set automatically according to (Algorithm 2 or Algorithm 3, [10]). Here, the condition we enforce is

$$\phi(\mathbf{z} + \alpha \tilde{\mathbf{d}}_p; \mu, \pi) \leq \phi(\mathbf{z}; \mu, \pi) - \eta_2 \alpha \Delta m_k(\mathbf{d}_p; \mu, \pi),$$

where $\eta_2 \in (0, 1)$ is a constant (we choose $\eta_2 = 10^{-8}$), and where Δm_k bounds the negative directional derivative of ϕ along $\tilde{\mathbf{d}}_p$ from below; see (2.6). Then we update the primal variables as

$$\mathbf{z} := \mathbf{z} + \alpha \tilde{\mathbf{d}}_p.$$

Finally for the dual variables, the step length β is set to the smallest value in $[\alpha, 1]$ that leads to a dual infeasibility reduction at least as large as a full Newton step, that is

$$(2.9) \quad \beta = \min \left\{ \tilde{\beta} \in [\alpha, 1] \mid \|(\mathbf{c}(\mathbf{z}, \boldsymbol{\lambda} + \tilde{\beta} \mathbf{d}_d))\| \leq \|(\mathbf{c}(\mathbf{z}, \boldsymbol{\lambda} + \mathbf{d}_d))\| \right\}.$$

The interior-point framework is summarized in Algorithm 1. In particular in Step 2, we check whether the current iterate is an infeasible stationary point, that is, whether it solves the problem

$$(2.10) \quad \min_{\mathbf{x} \in \mathbb{R}^n} \frac{1}{2} \sum_{k=1}^{N_E} \|\mathbf{A}_k(\mathbf{y}_k, \mathbf{u})\|^2 + \frac{1}{2} \|\max\left\{ \begin{pmatrix} \mathbf{u}_- - \mathbf{u} \\ \mathbf{u} - \mathbf{u}_+ \end{pmatrix}, \mathbf{0} \right\}\|^2,$$

where the maximum is meant componentwise. For any feasible PDE parameter \mathbf{u} , the last term in (2.10) is zero. Thus, a stationary point of (2.10) satisfies

$$(\mathbf{J}_{\mathbf{y}_k} \mathbf{J}_{\mathbf{u}})^\top \mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) = \mathbf{0}.$$

Since $\mathbf{J}_{\mathbf{y}_k}$ originates from a (well-posed) discretized and linearized PDE, it ought to be invertible; hence, $\mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) = \mathbf{0}$ and (\mathbf{y}, \mathbf{u}) is feasible. Therefore, the termination in Step 2 will never occur, as confirmed by our numerical experiments in Section 4.

3. Reduced-space preconditioning. At each Newton iteration, we must solve the very large but sparse linear system (2.5), where $\mathbf{H}_{\mathbf{y}\mathbf{y}}$ and $\mathbf{H}_{\mathbf{u}\mathbf{u}}$ are symmetric. Moreover, since the PDE constraints in (2.1) are only coupled through the control variable \mathbf{u} and the objective function \mathbf{F} is additive, the largest dominant block $\mathbf{H}_{\mathbf{y}\mathbf{y}}$ is in fact block diagonal,

$$(3.1) \quad \mathbf{H}_{\mathbf{y}\mathbf{y}} = \begin{pmatrix} \mathbf{V}^\top \mathbf{V} + \sum_i (\lambda_1)_i \nabla_{\mathbf{y}\mathbf{y}}^2 (\mathbf{A}_1)_i & & \\ & \ddots & \\ & & \mathbf{V}^\top \mathbf{V} + \sum_i (\lambda_{N_E})_i \nabla_{\mathbf{y}\mathbf{y}}^2 (\mathbf{A}_{N_E})_i \end{pmatrix},$$

where $(\mathbf{A}_k)_i$ is the i th component of $\mathbf{A}_k(\mathbf{y}_k, \mathbf{u})$. Similarly, the constraint Jacobian $\mathbf{J}_{\mathbf{y}}$ only consists of diagonal blocks,

$$(3.2) \quad \mathbf{J}_{\mathbf{y}} = \begin{pmatrix} \mathbf{J}_{\mathbf{y}_1} & & \\ & \ddots & \\ & & \mathbf{J}_{\mathbf{y}_{N_E}} \end{pmatrix}.$$

In fact if the partial differential operators $\mathcal{A}_k(y_k, u)$ in (1.1) are linear in y_k , that is if $\mathbf{A}_k(\mathbf{y}_k, \mathbf{u}) = \mathbf{A}_k(\mathbf{u})\mathbf{y}$, then each $\mathbf{J}_{\mathbf{y}_k} = \mathbf{A}_k(\mathbf{u})$. Moreover, when all the PDEs are identical, all the diagonal blocks in $\mathbf{H}_{\mathbf{y}\mathbf{y}}$ or $\mathbf{J}_{\mathbf{y}}$ coincide.

First, we permute the linear system (2.5) to swap the (2, 2) and (4, 4) blocks, and denote by \mathbf{D} the resulting matrix, which we rewrite as

$$\mathbf{D} = \begin{pmatrix} \mathbf{Q} & \mathbf{V} \\ \mathbf{V}^\top & \mathbf{P} \end{pmatrix},$$

with $\delta = 0$ for simplicity, where

$$\mathbf{Q} = \begin{pmatrix} \mathbf{H}_{\mathbf{y}\mathbf{y}} & \mathbf{J}_{\mathbf{y}}^\top \\ \mathbf{J}_{\mathbf{y}} & \mathbf{0} \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} \Sigma \mathbf{S}^{-1} \Lambda_{\mathcal{I}} \Sigma & \mathbf{0} & -\Sigma \\ \mathbf{0} & \mathbf{H}_{\mathbf{u}\mathbf{u}} & \hat{\mathbf{I}}^\top \\ -\Sigma & \hat{\mathbf{I}} & \mathbf{0} \end{pmatrix}, \quad \mathbf{V} = \begin{pmatrix} \mathbf{0} & \mathbf{H}_{\mathbf{y}\mathbf{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{\mathbf{u}} & \mathbf{0} \end{pmatrix}.$$

Typically the size of \mathbf{P} is much smaller than the size of \mathbf{Q} . Indeed a very large number of PDE parameters can lead to unphysical optimal solutions [28, 22]. Then it is particularly attractive to pursue the solution of the original linear system by forming and solving first the Schur-complement system with respect to the (2, 2) block \mathbf{P} of \mathbf{D} , given by

$$\mathbf{G} = \mathbf{P} - \mathbf{V}^\top \mathbf{Q}^{-1} \mathbf{V}.$$

The particular structure of \mathbf{Q} allows its inverse to be written explicitly as

$$(3.3) \quad \mathbf{Q}^{-1} = \begin{pmatrix} \mathbf{0} & \mathbf{J}_{\mathbf{y}}^{-1} \\ \mathbf{J}_{\mathbf{y}}^{-\top} & -\mathbf{J}_{\mathbf{y}}^{-\top} \mathbf{H}_{\mathbf{y}\mathbf{y}} \mathbf{J}_{\mathbf{y}}^{-1} \end{pmatrix}.$$

Therefore the solution of linear systems with system matrix \mathbf{Q} appears trivial provided that a robust and efficient solver for the inversion of $\mathbf{J}_{\mathbf{y}}$ is available.

Hence, let the (permuted) linear system take the form

$$\begin{pmatrix} \mathbf{Q} & \mathbf{V} \\ \mathbf{V}^\top & \mathbf{P} \end{pmatrix} \begin{pmatrix} \mathbf{d}_{y\lambda_\varepsilon} \\ \mathbf{d}_{us} \end{pmatrix} = \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix},$$

where $\mathbf{d}_{y\lambda_\varepsilon}^\top$ stands for the $(\mathbf{d}_y^\top \mathbf{d}_{\lambda_\varepsilon}^\top)$ and \mathbf{d}_{us}^\top for $(\mathbf{d}_s^\top, \mathbf{d}_u^\top, \mathbf{d}_{\lambda_z}^\top)$. First, we transform it to

$$\begin{pmatrix} \mathbf{Q} & \mathbf{V} \\ \mathbf{0} & \mathbf{P} - \mathbf{V}^\top \mathbf{Q}^{-1} \mathbf{V} \end{pmatrix} \begin{pmatrix} \mathbf{d}_{y\lambda_\varepsilon} \\ \mathbf{d}_{us} \end{pmatrix} = \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 - \mathbf{V}^\top \mathbf{Q}^{-1} \mathbf{w}_1 \end{pmatrix}$$

which we then solve by using the following algorithm. Clearly we never explicitly form the dense Schur-complement \mathbf{G} in the third step of Algorithm 2, but instead again use GMRES preconditioned by the matrix \mathbf{P} .

Algorithm 2 Reduced-Space Preconditioned GMRES Solver (RSP-GMRES)

- 1: Solve $\mathbf{Q}\mathbf{p}_1 = \mathbf{w}_1$
 - 2: Form $\mathbf{p}_2 := \mathbf{w}_2 - \mathbf{V}^\top \mathbf{p}_1$
 - 3: Solve $\mathbf{G}\mathbf{d}_{us} = \mathbf{p}_2$, where $\mathbf{G} = \mathbf{P} - \mathbf{V}^\top \mathbf{Q}^{-1} \mathbf{V}$
 - 4: Form $\mathbf{p}_1 := \mathbf{w}_1 - \mathbf{V}\mathbf{d}_{us}$
 - 5: Solve $\mathbf{Q}\mathbf{d}_{y\lambda_\varepsilon} = \mathbf{p}_1$
-

Since \mathbf{H}_{yy} and \mathbf{J}_y in (3.1)–(3.2) are both block-diagonal, the PDEs can be solved independently of one another and in parallel. Hence if every diagonal block is assigned to a separate processor, the total execution time becomes constant and independent of N_E . When the PDEs $A_k(y_k, u)$ are linear in y_k and the diagonal blocks \mathbf{J}_{y_k} , $k = 1, \dots, N_E$ all the same, just a single block needs to be factorized yielding linear run-time complexity with respect to N_E , even on a single processor.

4. Numerical Results. To demonstrate the effectiveness of our inexact interior-point (IIP) method, we shall now apply it to four different PDE-constrained optimization problems. In the first two examples from *optimal control*, the control variable u appears on the right-hand side of the PDE constraint and attempts to steer the system's state, y , towards a desired state \hat{y} . In the last two examples from *parameter estimation*, the spatially distributed model parameter u appears inside the differential operator itself, which typically leads to a nonconvex inverse problem. Here, additional knowledge about the true model \hat{u} , such as inequality constraints or multiple measurements, not only reduce the number of false local minima, but also mitigates the effect of noise in the observations \hat{y} .

In all examples, we follow the “discretize-then-optimize” approach, where the objective function and the PDE-constraints are first approximated numerically before applying our IIP method to the resulting finite-dimensional nonlinear optimization problem. To determine the new search direction at each Newton iteration, we apply to the (linear) KKT system (2.5) the reduced-space preconditioned (RSP) GMRES method, described in Section 3; in particular, prior to the GMRES iteration, the PDE block matrix \mathbf{J}_y in (3.3) is factorized.

Once the GMRES iterate meets the desired tolerance, it is evaluated by the SMART tests. If the termination tests (see Algorithms 2 and 3 in [10] for details) accept the new search direction, the optimization method proceeds with the step length computation, see Step 5 of Algorithm 1. If a Hessian modification is required, the Hessian block of the KKT system is modified as in (2.5) and the RSP-GMRES method is

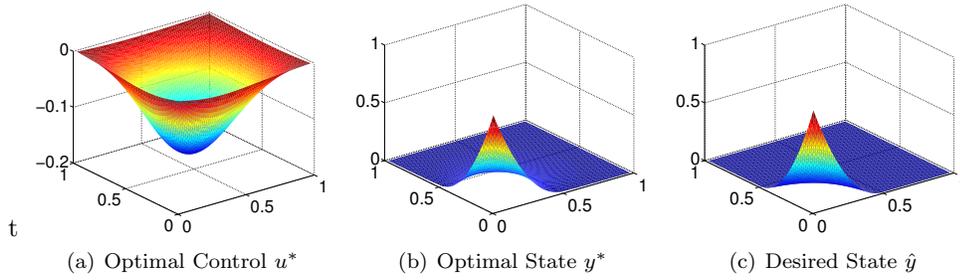


FIG. 1. *Corner load problem: Optimal control u^* , optimal state y^* and desired state \hat{y} with $N_h = 64$*

restarted. If none of the previous cases apply, the desired relative residual is reduced by a factor 10 while the RSP-GMRES iteration and SMART acceptance procedure are repeated. Due to (Lemma 3.2, [12]) this process will eventually terminate.

Both the IIP algorithm and the exact IP method, later used for the sake of comparison, have been implemented in the `Ipopt` open-source optimization package¹ (rev. 2094). Unless noted otherwise, we always set the first desired relative residual to 10^{-2} and use default parameter values elsewhere for the inexact algorithm. The RSP-preconditioner is implemented in C++ and uses a linear solver based on an *LU*-factorization from the `Pardiso` software package² (version 4.1.2.); again we use default settings unless noted otherwise. The IIP algorithm stops once the optimality conditions (2.3) are satisfied within a tolerance of 10^{-8} in the maximum norm.

All comparisons were performed on an Intel Xeon architecture with 128 GB main memory using Intel’s compiler version 10.1 under CentOS 5.8.

4.1. 2D Distributed Control. To illustrate the efficiency of our RSP-IIP approach, in particular with respect to the mesh size h , we now consider a PDE-constrained quadratic program, for which a multigrid based preconditioned projected conjugate gradient (PPCG) method was developed in [26]. Note that the PPCG algorithm requires a positive definite Hessian on the null-space of the Jacobian in (2.5); therefore, it cannot be applied to more difficult nonconvex problems, as we shall discuss in Sections 4.2–4.4.

Hence, we consider the following convex optimal control problem, either with a “corner load” or a “centered load” desired state \hat{y} :

$$\begin{aligned} \min_{y,u} F(y,u) &= \frac{1}{2} \|y - \hat{y}\|_{L^2(\Omega)}^2 + \frac{\alpha}{2} \|u\|_{L^2(\Omega)}^2, \\ \text{s.t. } -\Delta y &= u \quad \text{in } \Omega = (0,1)^2, \\ y &= g \quad \text{on } \partial\Omega, \end{aligned}$$

with $\alpha = 0.02$. For the *corner load* problem we set $g = \hat{y}|_{\partial\Omega}$, where

$$\hat{y}(x_1, x_2) = \begin{cases} (2x_1 - 1)^2(2x_2 - 1)^2 & \text{if } (x_1, x_2) \in [0, \frac{1}{2}]^2, \\ 0 & \text{otherwise,} \end{cases}$$

¹<http://www.coin-or.org/Ipopt/>

²<http://www.pardiso-project.org/>

TABLE 4.1

Corner load problem: Run-time [s] and iteration count (in parenthesis) for PPCG and RSP-GMRES, with varying tolerance tol and mesh size $N_h \times N_h$, leading to a KKT system of size n . All optimization problems were solved within a single (inexact) optimization step.

N_h	n	PPCG		RSP-GMRES	
		$tol = 10^{-6}$	$tol = 10^{-12}$	$tol = 10^{-6}$	$tol = 10^{-12}$
8	98	0.05 (3)	0.05 (4)	0.01 (3)	0.01 (5)
16	450	0.05 (2)	0.06 (4)	0.02 (3)	0.02 (5)
32	1'922	0.07 (2)	0.10 (4)	0.05 (3)	0.05 (5)
64	7'938	0.17 (1)	0.27 (3)	0.24 (3)	0.24 (5)
128	32'258	0.73 (1)	1.30 (3)	1.30 (3)	1.37 (5)
256	130'050	4.41 (1)	7.89 (3)	6.33 (3)	6.68 (5)
512	522'242	23.0 (1)	40.7 (3)	30.2 (3)	31.9 (5)

TABLE 4.2

Centered load problem: Run-time [s] and iteration count (in parenthesis) for PPCG and RSP-GMRES with varying tolerance tol and mesh size $N_h \times N_h$ leading to a KKT system of size n . All optimization problems were solved within a single inexact optimization step.

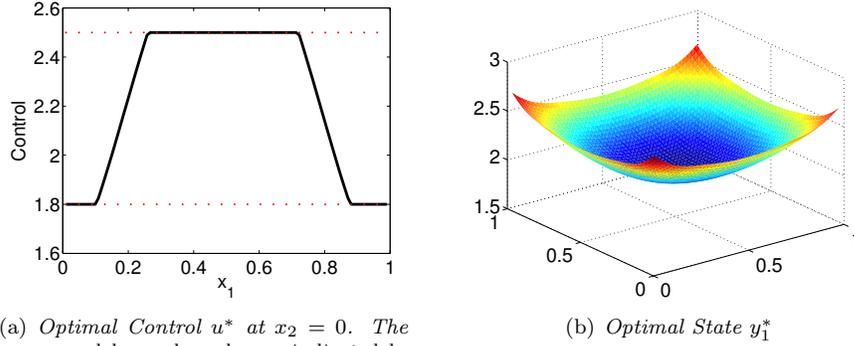
N_h	n	PPCG		RSP-GMRES	
		$tol = 10^{-6}$	$tol = 10^{-12}$	$tol = 10^{-6}$	$tol = 10^{-12}$
8	98	0.05 (2)	0.05 (3)	0.01 (3)	0.01 (5)
16	450	0.05 (2)	0.05 (3)	0.02 (3)	0.02 (5)
32	1'922	0.07 (2)	0.08 (3)	0.05 (3)	0.05 (5)
64	7'938	0.22 (2)	0.26 (3)	0.24 (3)	0.31 (5)
128	32'258	0.99 (2)	1.24 (3)	1.30 (3)	1.37 (5)
256	130'050	5.54 (2)	7.20 (3)	6.35 (3)	6.70 (5)
512	522'242	27.2 (2)	35.7 (3)	30.3 (3)	31.8 (5)

whereas for the *centered load* problem we let $g = 0$ and

$$\hat{y}(x_1, x_2) = \exp\left(-\frac{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}{0.125^2}\right).$$

The state and control variables y, u are discretized on a regular $N_h \times N_h$ grid with Q_1 finite elements. Next, we apply the RSP-IIP method with $\mu_{init} = 10^{-11}$ to both problems and compare run-times and iteration counts with those from the MATLAB implementation of the PPCG algorithm in [26]. The KKT systems (2.5) are solved iteratively either with PPCG or RSP-GMRES until the relative residual has reached a desired tolerance $tol = 10^{-6}$ or 10^{-12} . Regardless of problem size, the IIP algorithm, though inexact, always found a solution within a single optimization step, thus demonstrating the remarkable accuracy of its search direction.

In Tables 4.1 and 4.2, we compare run-times and iteration counts (in parenthesis) of the PPCG and the RSP-IIP method to solve the KKT system for varying mesh size and tolerance. In all cases, the number of iterations remains independent of N_h ; hence, our RS preconditioner also exhibits optimal h -independent behavior. The run-times of both algorithms are comparable, while the iteration counts differ by at most two. Recall, however, that the PPCG method cannot be applied to more general nonconvex problems, possibly with inequality constraints, as we shall discuss in the sequel.



(a) Optimal Control u^* at $x_2 = 0$. The upper and lower bounds are indicated by the dotted lines.

(b) Optimal State y_1^*

FIG. 2. 2D boundary control: Optimal state and control for $N_E = 1$ PDE constraint discretized on a $N_h \times N_h$ mesh with $N_h = 100$.

4.2. 2D Boundary Control Problem. Next, to demonstrate the robustness of the RSP-IIP method with respect to increasing mesh size and number of PDE constraints, we consider a 2D boundary control problem (example 5.7, [23]) with multiple nonlinear PDEs and inequality constraints. Since the PCG algorithm no longer applies here, we shall compare the RSP-IIP method to the exact IP method from [33] instead, where the KKT systems are solved by the sparse direct solver *Pardiso* [29]. By exploiting the sparsity structure of the KKT system, the RSP-IIP method achieves a significant speed-up over the exact IP method, even more so as the KKT system increases.

Hence, we consider the optimal boundary control problem:

$$(4.1a) \quad \min_{y,u} F(y,u) = \frac{1}{2N_E} \sum_{k=1}^{N_E} \|y_k - \hat{y}_k\|_{L^2(\Omega)}^2 + \frac{\alpha}{2} \|u\|_{L^2(\partial\Omega)}^2,$$

$$(4.1b) \quad \text{s.t. } -\Delta y_k - y_k + y_k^3 = 0 \quad \text{in } \Omega = (0,1)^2,$$

$$\frac{\partial y_k}{\partial n} = ku \quad \text{on } \partial\Omega = (0,1)^2,$$

$$(4.1b) \quad 1.8 \leq u \leq 2.5,$$

where

$$\hat{y}_k = k(2 - x_1(x_1 - 1) + x_2(x_2 - 1)), \quad k = 1, \dots, N_E,$$

and $\alpha = 0.01$. Here (4.1a) describes the interaction of normalized quantum mechanical wave functions of electrons y_k in a superconductor on the basis of a simplified Ginzburg-Landau model [32]. For each $k = 1, \dots, N_E$ the corresponding PDE constraint (4.1a) is discretized with standard second-order finite-differences on a regular $N_h \times N_h$ grid.

In Figure 2, the optimal control u^* and optimal state y^* are shown for a single PDE constraint, that is $N_E = 1$ and $N_h = 100$. Both are initialized as $u \equiv 2.15$ and $y = \hat{y}_k$, respectively. Note that the bounds (4.1b) on u are partially active and the KKT system thus becomes increasingly ill-conditioned as $\mu \rightarrow 0$ because of the barrier term in (2.2).

Next, in Figure 3, we show the number of RSP-GMRES iterations (\times) and the relative residual of each iterate accepted by the SMART test (\diamond) for varying N_E and

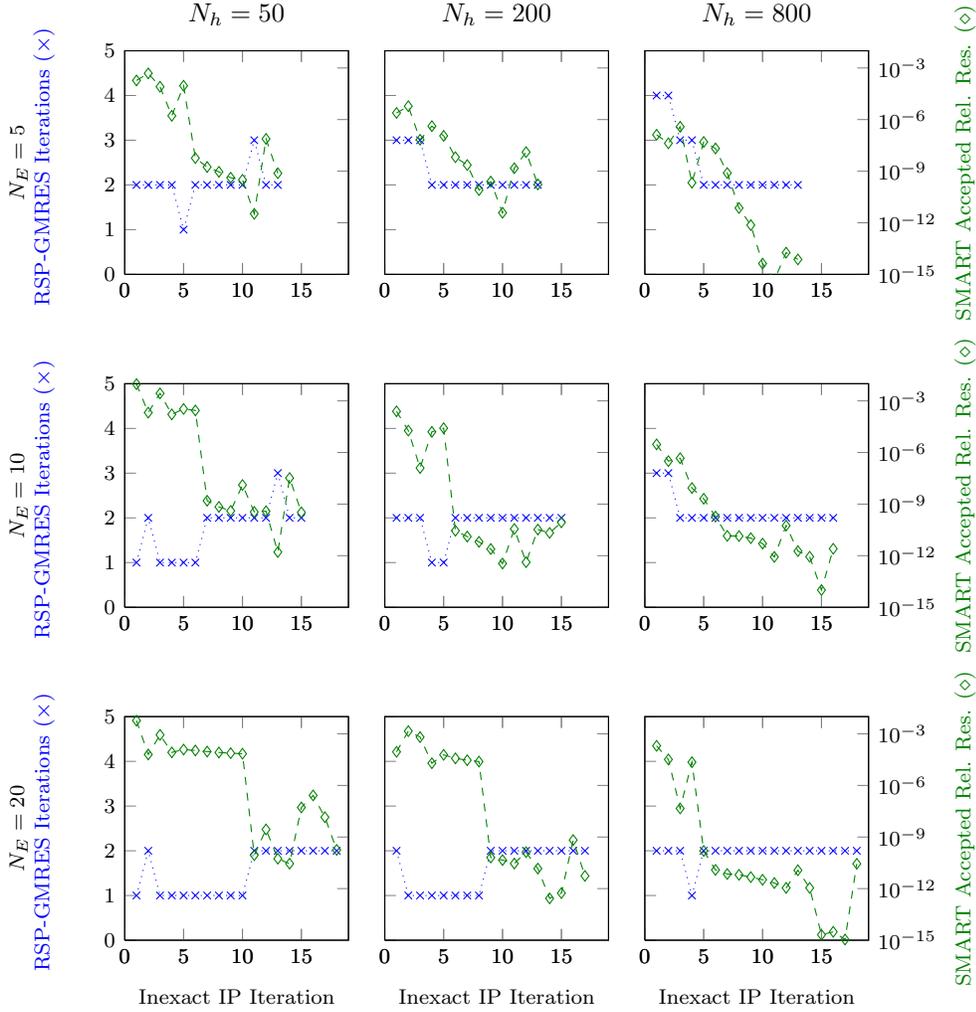


FIG. 3. 2D boundary control: number of RSP-GMRES iterations (\times) and relative residual (\diamond) at SMART acceptance for varying number of PDE constraints, N_E , and mesh sizes $N_h \times N_h$. Note the different scales on the left and right axes.

N_h . Independently of N_E or the mesh size, $N_h \times N_h$, the RSP-IIP method always converges within 20 optimization steps, while the RSP-GMRES iterates satisfy the SMART test within only five iterations.

In Table 4.3 we list for varying N_h and N_E the size of the resulting KKT system n , the total run-time, and the number of optimization steps for the RSP-IIP and the exact IP method. The number of optimization steps needed by the RSP-IIP method exceeds at most by four that of the exact IP method; it barely increases as the KKT system grows one thousand fold. By taking advantage of the KKT system's sparsity structure, the RSP-IIP method achieves a speedup of 36, and even more so with growing problem size. As the run-time per inexact interior-point iteration scales linearly with respect to the number of PDE constraints, N_E , while the number of optimization steps barely increases, we obtain essentially linear run-time complexity. Moreover, if the different diagonal blocks in \mathbf{J}_y and \mathbf{J}_y^\top in (3.2), all independent of

TABLE 4.3

2D boundary control problem: size of the KKT system, n , number of optimization steps and total run-time for our RSP-IIP method and an exact IP algorithm for varying numbers of PDE constraints, N_E , and mesh size $N_h \times N_h$.

Problem size			RSP-IIP		exact IP	
N_h	N_E	n	# IIP steps	run-time [s]	# IP steps	run-time [s]
50	5	26'000	13	3	12	3
50	10	51'000	15	7	13	10
50	20	101'000	18	17	14	10
200	5	404'000	13	78	12	125
200	10	804'000	15	173	13	697
200	20	1'604'000	17	382	14	3'829
800	5	6'416'000	13	1'661	13	10'479
800	10	12'816'000	16	4'106	14	68'750
800	20	25'616'000	18	9'132	14	335'470

one another, were solved in parallel, the total execution time would essentially become constant and thus independent of N_E .

4.3. 2D and 3D Parameter Estimation. We now consider two problems from groundwater modeling [14], the first in two and the second in three space dimensions, where the log conductivity, $u(x)$, needs to be estimated from noisy measurements of the fluid pressure, $y(x)$. Since the influence of noise can be further reduced by including measurements from multiple sources, practical applications often lead to multiple PDE constraints. Then, the (optimal) linear complexity achieved by the RSP-IIP algorithm with respect to the number of PDE constraints, N_E , becomes a key ingredient for its efficiency.

Thus, we let $\Omega = (0, 1)^d$, $d = 2, 3$, and consider the spatially distributed parameter estimation problem:

$$(4.2) \quad \begin{aligned} \min_{y, u} F(y, u) &= \frac{1}{2N_E 64} \sum_{k=1}^{N_E} \|v(y_k) - \hat{y}_k\|_{\ell^2}^2 + \frac{\alpha}{2} \left(V(u) + \beta \|u\|_{L^2(\Omega)}^2 \right), \\ \text{s. t. } -\nabla \cdot (\exp(u) \nabla y_k) &= q_k \quad \text{in } \Omega, k = 1, \dots, N_E \\ y_k &= 0 \quad \text{on } \partial\Omega, \\ -2 &\leq u \leq 2. \end{aligned}$$

Here, the true log conductivity is

$$\hat{u}(x) = \exp\left(-\frac{\|\tilde{x}_a - x\|^2}{0.05}\right) - \exp\left(-\frac{\|\tilde{x}_b - x\|^2}{0.05}\right),$$

with $\tilde{x}_a = (0.25, 0.25)^\top$, $\tilde{x}_b = (0.75, 0.75)^\top$ in two, or $\tilde{x}_a = (0.25, 0.25, 0.25)^\top$, $\tilde{x}_b = (0.75, 0.75, 0.75)^\top$ in three dimensions, respectively. To generate the synthetic measurements \hat{y}_k , $k = 1, \dots, N_E$, we numerically solve the forward problem for a given source, q_k , and add 1% of white noise to its solution.

In 2D, the source terms are given by

$$\begin{aligned} q_{4l+1}(x) &= \sin(\nu_1(l) 2\pi x_1) \sin(\nu_2(l) 2\pi x_2), \\ q_{4l+2}(x) &= \cos(\nu_1(l) 2\pi x_1) \sin(\nu_2(l) 2\pi x_2), \\ q_{4l+3}(x) &= \sin(\nu_1(l) 2\pi x_1) \cos(\nu_2(l) 2\pi x_2), \\ q_{4l+4}(x) &= \cos(\nu_1(l) 2\pi x_1) \cos(\nu_2(l) 2\pi x_2), \end{aligned}$$

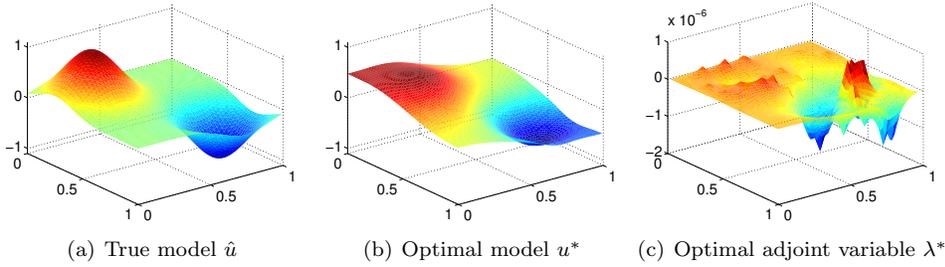


FIG. 4. 2D log conductivity estimation: true model and optimal solution with $N_E = 10$ PDE constraints on a $N_h \times N_h$ mesh with $N_h = 100$

TABLE 4.4

2D log conductivity estimation: size of the KKT system, n , number of optimization steps and total run-time for the RSP-IIP method for varying number of PDE constraints, N_E , and mesh size $N_h \times N_h$

N_h	N_E	n	# IIP steps	run-time [s]
100	5	151'005	9	25
100	10	251'005	9	38
100	20	451'005	9	60
200	5	602'005	9	109
200	10	1'002'005	9	155
200	20	1'802'005	9	256
400	5	2'404'005	11	579
400	10	4'004'005	9	649
400	20	7'204'005	9	1'073

with $(\nu_1(1), \nu_2(1)) = (1, 1)$, $(\nu_1(2), \nu_2(2)) = (1, 2)$, $(\nu_1(3), \nu_2(3)) = (2, 1)$, $(\nu_1(4), \nu_2(4)) = (1, 3)$, $(\nu_1(5), \nu_2(5)) = (2, 2), \dots$ and similarly in 3D by triple products of trigonometric functions. For each k , the corresponding pressure field $y_k(x)$ is measured at 64 fixed locations irregularly distributed throughout Ω ; those measurements are then collected in the vector $v(y_k)$.

In (4.2) each elliptic PDE is discretized on a regular grid using Q^0 finite elements for the model variable, u , and Q^1 finite elements for the state variable y_k . As a measure of variation in the piecewise constant log conductivity coefficient, we include the penalty term

$$V(u) = h^{2d} \int_{\mathcal{F}_h} \llbracket u \rrbracket^2 ds,$$

where \mathcal{F}_h denotes all inter-element boundaries and $\llbracket u \rrbracket$ denotes jumps across interfaces.

First, we consider the 2D case and apply our RSP-IIP method to (4.2) with $\alpha = 10^{-7}$ and $\beta = 0$. As initial guess, we always set u identically to zero and y_k to the corresponding forward solutions. Figure 4 shows the optimal solution for $N_E = 10$ PDE constraints on an $N_h \times N_h$ mesh with $N_h = 100$. Indeed, the optimal model approximately follows the true model, while the measurements' misfits act as point sources in the adjoint problem for the Lagrange multiplier λ . In Table 4.4 we list the number of optimization steps and total run-time for varying mesh size $N_h \times N_h$ and number of PDE constraints, N_E , thereby resulting in KKT systems up to $n = 7.2 \cdot 10^6$

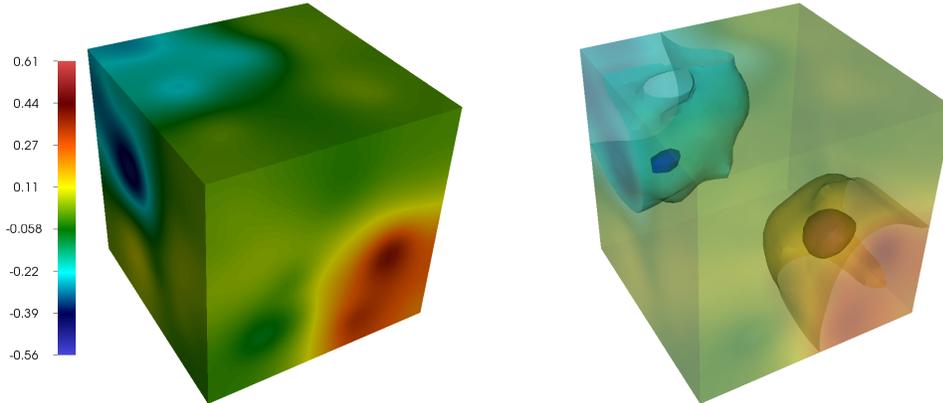


FIG. 5. 3D log conductivity estimation: Optimal model u^* computed by the RSP-IIP method applied to (4.2) with $N_E = 6$ PDE constraints on a $25 \times 25 \times 25$ grid (left) and corresponding iso-surfaces at $[-0.45, -0.2, 0.2, 0.45]$ (right).

TABLE 4.5

3D log conductivity estimation: size of KKT system, n , number of optimization steps and total run-time for the RSP-IIP method for varying number of PDE constraints, N_E , and mesh size N_h^3

N_h	N_E	n	# IIP steps	run-time [s]
20	4	110'305	11	59
20	8	174'305	11	80
20	16	302'305	11	129
20	32	558'305	9	202
40	4	856'605	15	1'452
40	8	1'368'605	12	1'345
40	16	2'392'605	12	1'913
40	32	4'440'605	12	3'102
60	4	2'862'905	15	10'180
60	8	4'590'905	14	11'491
60	16	8'046'905	14	14'331
60	32	14'958'905	13	18'595

in size. Remarkably, every problem was solved within merely 11 optimization steps, independently of N_h and N_E , while the total run-time increases only linearly with N_E . Since the PDE in (4.2) is linear in the state variable y , as is often the case, the sub-matrices \mathbf{J}_{y_k} in (3.2) are all identical; hence, the RSP-GMRES solver only needs to factorize a single PDE block-matrix, such as \mathbf{J}_{y_1} .

Next, we consider the 3D case, and apply the RSP-IIP method to (4.2) with $\alpha = 10^{-7}$ and $\beta = 1$. Figure 5 shows slices of a typical optimal model u^* for $N_E = 6$ PDE constraints on an $N_h \times N_h \times N_h$ mesh with $N_h = 25$, which also exhibits a maximum about $(0.25, 0.25, 0.25)$ and a minimum about $(0.75, 0.75, 0.75)$.

In Table 4.5 we list the number of optimization steps and total run-time for varying N_E and mesh sizes $N_h \times N_h \times N_h$. Again, all problems were solved within merely 15 optimization steps, which barely increases with growing mesh size. Moreover, the RSP-IIP method efficiently takes advantage of the added information from

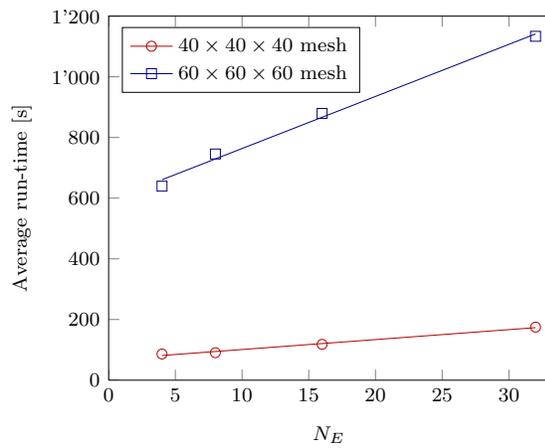


FIG. 6. *3D log conductivity estimation: average run-time per optimization step of the RSP-IIP method as a function of the number of PDE constraints, N_E , for an N_h^3 mesh with $N_h = 40, 60$.*

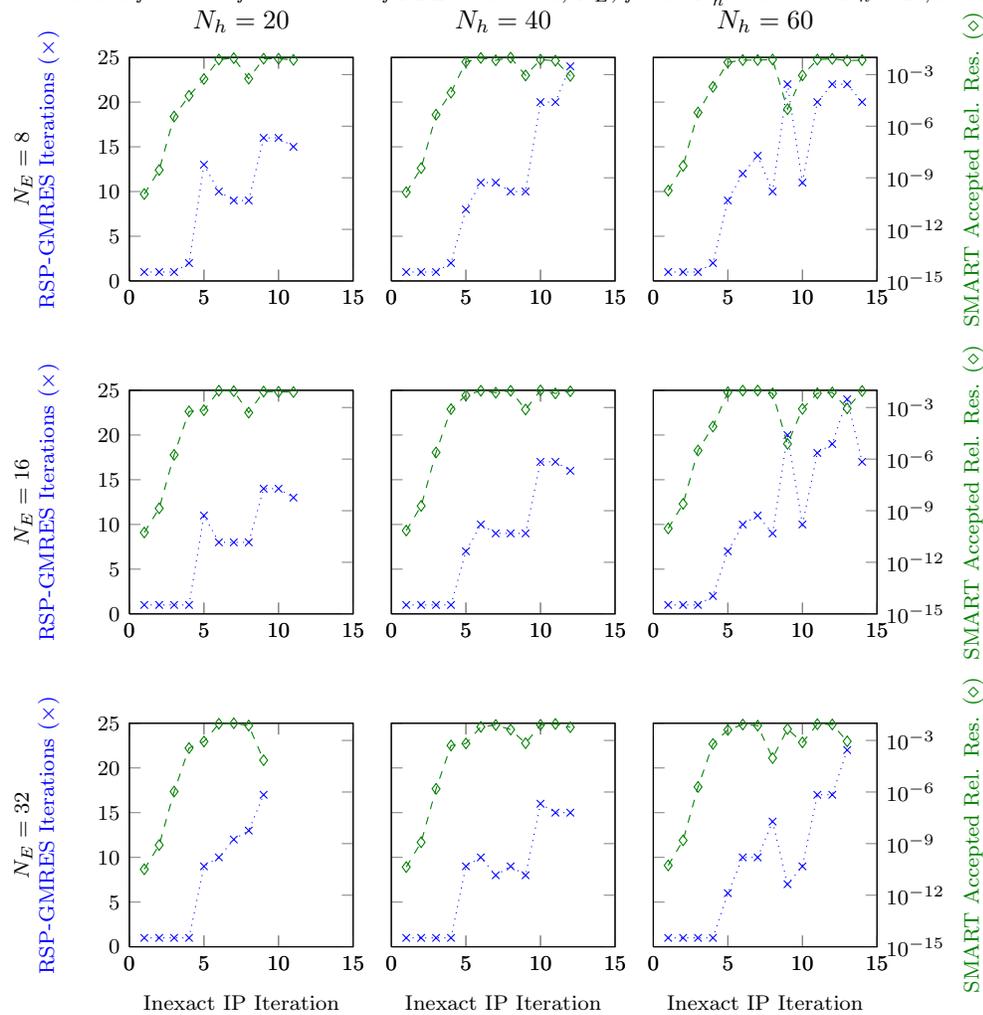


FIG. 7. *3D log conductivity estimation: number of RSP-GMRES iterations (\times) and relative residual (\diamond) at SMART acceptance for varying number of PDE constraints, N_E , and mesh sizes $N_h \times N_h \times N_h$. Note the different scales on the left and right axes.*

increasingly many PDE constraints, as the number of optimization steps actually decreases for increasing N_E . In Figure 6, we again observe the linear average run-time complexity of the RSP-IIP method with respect to N_E for two different $N_h \times N_h \times N_h$ meshes. Remarkably, the total run-time is even sub-linear due to the mild decrease in the number of optimization steps with increasing N_E . In Fig. 7 we also follow the evolution of the number of RSP-GMRES iterations (' \times ') and the relative residual of the accepted search direction (' \diamond ') at each optimization step. The relative residual of the accepted iterate, never below 10^{-6} , typically lies above 10^{-2} . Hence, the SMART tests do not demand increasingly accurate solutions from the RSP-GMRES solver during the optimization process. Although the number of RSP-GMRES iterations typically increases in the course of any optimization run, all KKT systems were solved within 25 RSP-GMRES iterations, independently of N_h or N_E .

4.4. Geophysical Imaging. Full waveform seismic inversion leads to some of the most challenging nonlinear PDE-constrained optimization problems. Here we consider the Marmousi model [8, 31, 34], a standard benchmark in seismic imaging. It corresponds to a vertical slice through the Cuanza basin in Angola [19] delimited by the two dimensional domain $\Omega = (0.4 \text{ km}, 6.4 \text{ km}) \times (0 \text{ km}, 1.6 \text{ km})$.

Given measurements, $\hat{y}_k, k = 1, \dots, N_E$ from $N_E = 11$ seismic events ("shots"), we shall attempt to reconstruct the true velocity profile $\hat{u}(x)$ in Ω – see Figure 8 (a). The measurements are obtained for each point source δ_k by recording the amplitude of the corresponding (complex-valued) wave field, y_k , at 367 observation points located at the surface and the two lateral boundaries (vertical wells); those values are then collected in the vector $v(y_k)$. Hence, we consider the PDE-constrained optimization problem:

$$(4.3) \quad \begin{aligned} \min_{y, u} F(y, u) &= \frac{1}{2} \sum_{k=1}^{N_E} \|v(y_k) - \hat{y}_k\|_{\ell^2}^2 + \frac{\alpha}{2} TV(u), \\ \text{s.t. } -\nabla \cdot (u^2 \nabla y_k) - \omega^2 y_k &= \delta_k \quad \text{in } \Omega, \\ \frac{\partial y_k}{\partial n} &= i \frac{\omega}{u} y_k \quad \text{on } \partial\Omega, \\ u_- &\leq u \leq u_+. \end{aligned}$$

Here $TV(u) = \int_{\Omega} \sqrt{|\nabla u|^2 + \varepsilon} dx$ denotes regularization by total variation with $\varepsilon = 18.75$ and $\alpha = 10^{-4}$. Each wave field, y_k , satisfies the Helmholtz equation (forward problem) for a given velocity profile $u(x)$ and frequency $\omega > 0$. At the boundary of the computational domain, $\partial\Omega$, we impose a first-order Sommerfeld-like absorbing boundary condition. To generate the synthetic data \hat{y}_k , we first solve the forward problem on a fine mesh with mesh size $h = 8 \text{ m}$ using Q^1 elements for u and Q^2 elements for y_k ; then we add 1% white noise to the recorded values. The finite element discretization was implemented using `libMesh` [21] and `PETSc` [1].

Since the problem is nonconvex, any local optimization method may converge to a false solution. Prior knowledge, in practice often available as inequality constraints, reduces the search space and thereby prevents the algorithm from becoming trapped too easily in a (false) local minimum. Here, the upper and lower bounds u_+, u_- are determined from \hat{u} by local mean and maximum, or minimum filtering [30] respectively, followed by 10% further expansion of the feasible interval – see Figure 8 (c) and (d).

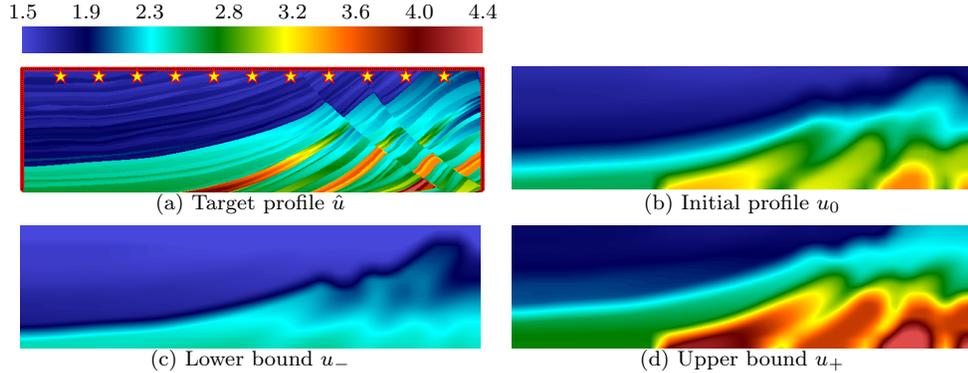


FIG. 8. *Geophysical imaging: target velocity profile in (km/s) of the Marmousi benchmark \hat{u} (a), and the initial guess u_0 (b) that will be used for its reconstruction, along with the upper (c) and lower (d) bounds for the control u_+, u_- . The stars indicate the location of the different point sources δ_k .*

As the number of extrema increases with frequency, we further reduce the risk of ending up in a (false) local minimum through frequency stepping: starting at lower frequency ω , we progressively increase ω while initializing each optimization run from the previous optimal model obtained at lower frequency. In all cases we set $\varepsilon_{tol} = 10^{-4}$ in (2.4), as smaller values did not yield any further improvement in the reconstruction. Moreover, we discretize both, u and y_k with Q^1 finite elements on a coarser mesh, thus avoiding any potential “inverse crime”.

Starting at the lowest frequency, $\omega = 20$, we now apply the RSP-IIP method to (4.3) with mesh size $h = 40$ m for u and $h = 20$ m for y_k . We initialize the algorithm with $\mu = 10^{-3}$ and $u = (u_+ + u_-)/2$ – see Figure 8. After only 10 optimization steps, the RSP-IIP method converges to the optimal model u^* , displayed in Figure 9; the imaginary part of one typical wave field, y_2 is also shown there. At such low frequency, the wave length is still large and hence unable to detect smaller features in the medium – see also Figure 10. The total run-time and problem size are summarized in Table 4.6. In Figure 11, we follow the number of RSP-GMRES iterations required to solve the $n \times n$ system with $n = 1.1 \cdot 10^6$.

Next, we let $\omega = 40$ and initialize the RSP-IIP algorithm with the optimal point from the previous run at $\omega = 20$. Again, we observe in Figure 11 an increase in the number of RSP-GMRES iterations during the optimization process. Although the problem size and the number of optimization steps are identical, the overall larger number of RSP-GMRES iterations results in a slight increase in total run-time – see Table 4.6.

Finally, we let $\omega = 60$, $\mu = 10^{-5}$, and choose a finer mesh width $h = 20$ m for the model parameter, too. As a consequence, the KKT system barely increases, whereas the KKT Schur complement now increases by a factor four. Nonetheless, the RSP-IIP method converges in only 14 steps, while the number of RSP-GMRES iterations never exceeds 300 iterations. The reconstructed model, shown in Figure 9, now reproduces the sharp transitions in the velocity field from the true model, with even smallest details revealed in Figure 10.

For comparison we now also apply the exact IP method to (4.3) with $\omega = 40$. The linear systems are solved using the direct solver `Pardiso` with 4 OpenMP threads.

TABLE 4.6

Geophysical imaging: time frequency, ω , problem size, size of the KKT system, n , number of inexact interior-point iterations, and run-times.

ω	# States	# Model Param.	n	# IIP steps	run-time [s]
20	$5.36 \cdot 10^5$	$6.19 \cdot 10^3$	$1.10 \cdot 10^6$	10	825
40	$5.36 \cdot 10^5$	$6.19 \cdot 10^3$	$1.10 \cdot 10^6$	10	1'001
60	$5.36 \cdot 10^5$	$2.44 \cdot 10^4$	$1.19 \cdot 10^6$	14	1'734

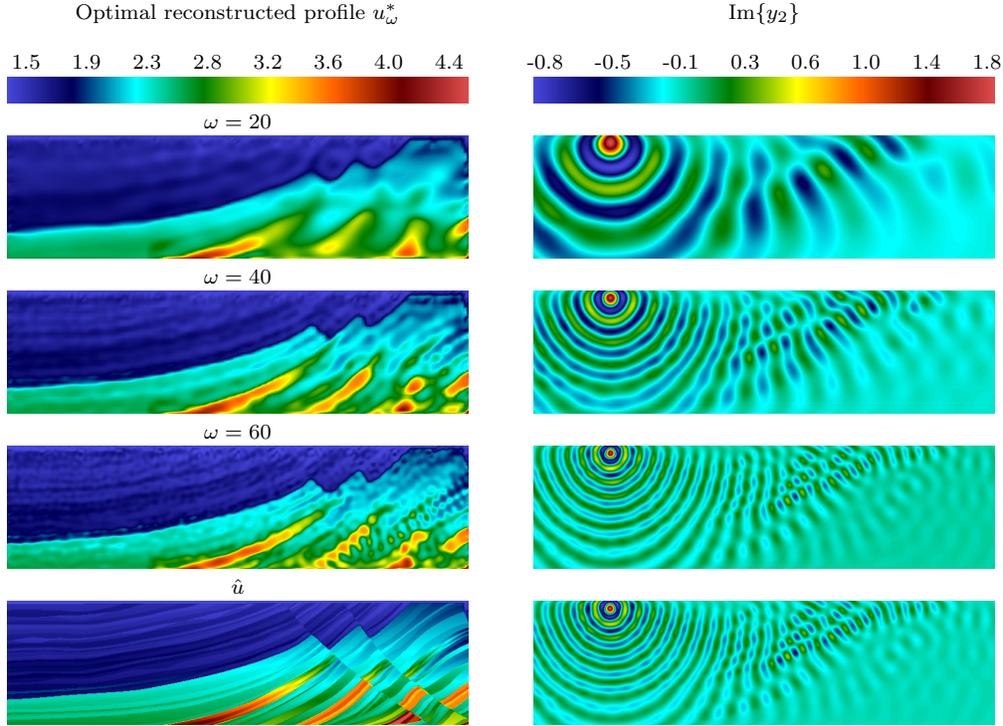


FIG. 9. Geophysical imaging: initial model for $\omega = 20$, and reconstructed optimal model for $\omega = 20, 40$, and 60 (left); Imaginary part of the y_2 for $\omega = 20, 40$, and 60 (right). Compare with true model in Figure 8.

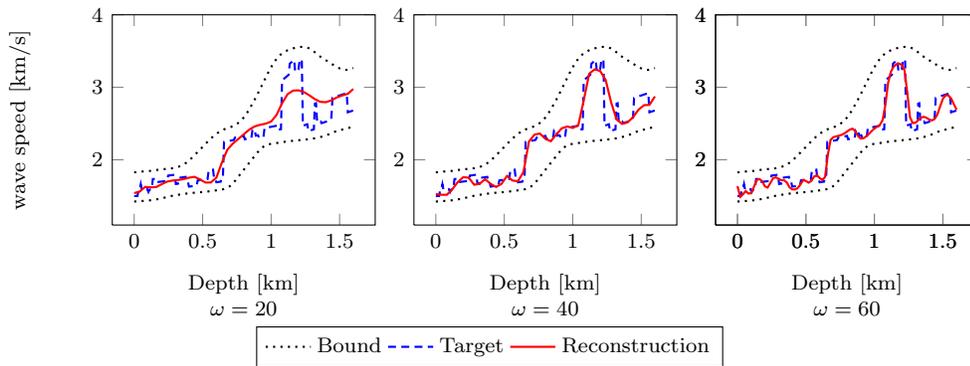


FIG. 10. Geophysical imaging: vertical cross-section of the true wave speed \hat{u} , its reconstruction u^* and lower and upper bounds u_+ and u_- at $x_1 = 4.0$ at increasingly higher frequencies $\omega = 20, 40$ and 60 .

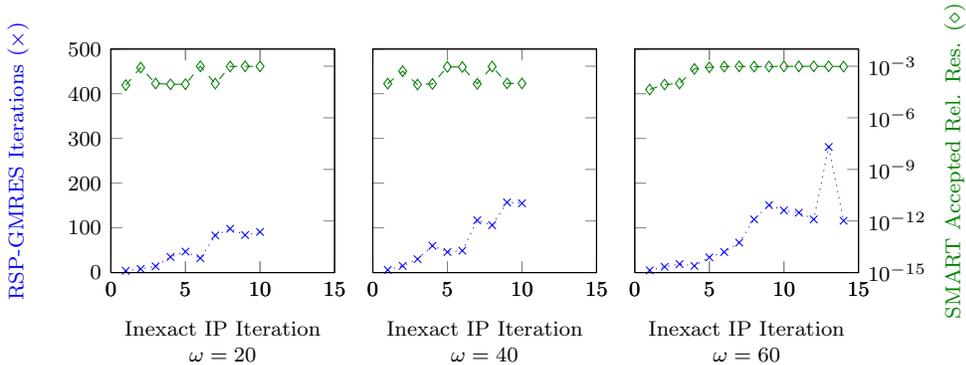


FIG. 11. *Geophysical imaging: number of RSP-GMRES iterations (\times) and relative residual (\diamond) at SMART acceptance for $\omega = 20, 40$ and 60 . Note the different scales on the left and right axes.*

The exact IP method converges in 9 optimization steps, instead of 10 for the inexact IP method, yet its total run-time of 14'993s now results in a fifteenth fold increase, despite the multi-threaded hardware! This illustrates that the SMART tests efficiently control the inexactness, thus leading to very few additional optimization steps, while the RSP-GMRES solver takes advantage of the sparsity structure of the KKT systems.

5. Concluding Remarks. Starting from the inexact interior-point (IIP) framework from [10, 12], we have devised an effective reduced-space preconditioner (RSP) for the iterative solution of the KKT system needed at each Newton step. Together they yield a scalable and robust numerical method for the solution of large-scale non-convex PDE-constrained optimization problems with inequality constraints. Because it uses the full Hessian matrix, modifying it whenever needed, the RSP-IIP method is not only globally convergent, but also converges fast locally.

The RSP preconditioner is not tailored to any particular class of PDEs or constraints, but instead judiciously exploits the structure of the KKT system. It efficiently handles any number of PDE constraints, N_E , a key advantage for inverse problems where the influence of noisy data is often reduced by including data from multiple sources. Moreover, our numerical experiments show that the number of optimization steps remains independent of N_E . Similarly, both the number of inner RSP-GMRES iterations and of outer Newton iterations remains essentially independent of the mesh size h . As a consequence, we repeatedly observe (optimal) linear run-time complexity with respect to N_E or to h , even in the presence of active inequality constraints.

The number of optimization steps needed by the RSP-IIP method is always comparable to that of the corresponding exact IP approach, where the KKT system is explicitly factorized. Yet by exploiting the sparsity structure of the KKT system (2.5), the RSP-IIP method achieves a significant speed-up over the exact IP method, even more so as the size of the KKT system increases. Since the two dominant blocks of the KKT system (2.5) are both block-diagonal, the PDEs can be solved independently of one another and in parallel. Hence if every forward solve is assigned to a separate processor, the total execution time becomes independent of N_E .

Acknowledgement. The authors would like to thank Tyronne Rees for the PCG Matlab code used in Section 5.1, and Ahmed Sameh, Saied Faisal and Alicia Klinxev for useful comments and suggestions.

REFERENCES

- [1] S. BALAY, J. BROWN, K. BUSCHELMAN, W. D. GROPP, D. KAUSHIK, M. G. KNEPLEY, L. C. MCINNES, B. F. SMITH, AND H. ZHANG, *PETSc Web page*, 2013. <http://www.mcs.anl.gov/petsc>.
- [2] M. BENZI, E. HABER, AND L. TARALLI, *Multilevel algorithms for large-scale interior point methods*, SIAM J. Sci. Comput., 31 (2009), pp. 4152–4175.
- [3] ———, *A preconditioning technique for a class of PDE-constrained optimization problems*, Adv. Comput. Math., 35 (2011), pp. 149–173.
- [4] G. BIROS AND O. GHATTAS, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part I: The Krylov-Schur solver*, SIAM J. Sci. Comput., 27 (2005), pp. 687–713.
- [5] ———, *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 J. Sci. Comput., 27 (2005), pp. 714–739.
- [6] M. BOLLHÖFER, M. J. GROTE, AND O. SCHENK, *Algebraic multilevel preconditioner for the Helmholtz equation in heterogeneous media*, SIAM J. Sci. Comput., 31 (2009), pp. 3781–3805.
- [7] A. BORZI AND V. SCHULZ, *Multigrid methods for PDE optimization*, SIAM Rev., 51 (2009), pp. 361–395.
- [8] C. BUNKS, F. M. SALECK, S. ZALESKI, AND G. CHAVENT, *Multiscale seismic waveform inversion*, Geophysics, 60 (1995), pp. 1457–1473.
- [9] R. BYRD, F. E. CURTIS, AND J. NOCEDAL, *An inexact Newton method for nonconvex equality constrained optimization*, Math. Program., 122 (2010), pp. 273–299. 10.1007/s10107-008-0248-3.
- [10] F. E. CURTIS, J. HUBER, O. SCHENK, AND A. WÄCHTER, *A note on the implementation of an interior-point algorithm for nonlinear optimization with inexact step computations*, Math. Program., (2012), pp. 1–19.
- [11] F. E. CURTIS, J. NOCEDAL, AND A. WÄCHTER, *A matrix-free algorithm for equality constrained optimization problems with rank-deficient jacobians*, SIAM J. Optim., 20 (2009), pp. 1224–1249.
- [12] F. E. CURTIS, O. SCHENK, AND A. WÄCHTER, *An interior-point algorithm for large-scale nonlinear optimization with inexact step computations*, SIAM J. Sci. Comput., 32 (2010), pp. 3447–3475.
- [13] E. HABER AND U. M. ASCHER, *Preconditioned all-at-once methods for large, sparse parameter estimation problems*, Inverse Problems, 17 (2001), pp. 1847–1864.
- [14] E. HABER AND L. HANSON, *Model problems in PDE-constrained optimization*, tech. report, Emory University, 2007.
- [15] M. HEINKENSCHLOSS AND D. RIDZAL, *An inexact trust-region SQP method with applications to pde-constrained optimization*, in Numerical Mathematics and Advanced Applications: Proceedings of ENUMATH 2007, the 7th European Conference on Numerical Mathematics and Advanced Applications, Graz, Austria, K. Kunisch, G. Of, and O. Steinbach, eds., Springer, 2008, pp. 613–620.
- [16] M. HEINKENSCHLOSS AND L. N. VICENTE, *Analysis of inexact trust-region SQP algorithms*, SIAM J. Optim., 12 (2002), pp. 283–302.
- [17] R. HERZOG AND K. KUNISCH, *Algorithms for PDE-constrained optimization*, GAMM-Mitteilungen, 33 (2010), pp. 163–176.
- [18] M. HINZE, R. PINNAU, M. ULBRICH, AND S. ULBRICH, *Optimization with PDE Constraints*, vol. 23 of Mathematical Modeling: Theory and Applications, Springer, Dordrecht, Netherlands, 2009.
- [19] T. IRONS, *Marmousi model*, 2007. http://www.reproducibility.org/RSF/book/data/marmousi/paper_html/node1.html.
- [20] H. JÄGER AND E. W. SACHS, *Global convergence of inexact reduced SQP methods*, Optimization Methods and Software, 7 (1997), pp. 83–110.
- [21] B. S. KIRK, J. W. PETERSON, R. H. STOGNER, AND G. F. CAREY, *libMesh: A c++ library for parallel adaptive mesh refinement/coarsening simulations*, Engineering with Computers, 22 (2006), pp. 237–254. <http://dx.doi.org/10.1007/s00366-006-0049-3>.

- [22] X. MA AND N. ZABARAS, *Kernel principal component analysis for stochastic input model generation*, Journal of Computational Physics, 230 (2011), pp. 7311–7331.
- [23] H. MAURER AND H. D. MITTELMANN, *Optimization techniques for solving elliptic control problems with control and state constraints: Part 1. boundary control*, Computational Optimization and Applications, 16 (2000), pp. 29–55. 10.1023/A:1008725519350.
- [24] J. NOCEDAL AND S. WRIGHT, *Numerical Optimization*, Operations Research and Financial Engineering, Springer, USA, 2006.
- [25] E. E. PRUDENCIO, R. H. BYRD, AND X.-C. CAI, *Parallel full space SQP Lagrange-Newton-Krylov-Schwarz algorithms for PDE-constrained optimization problems*, SIAM J. Sci. Comput., 27 (2005), pp. 1305–1328.
- [26] T. REES, H. S. DOLLAR, AND A. J. WATHEN, *Optimal solvers for PDE-constrained optimization*, SIAM J. Sci. Comput., 32 (2010), pp. 271–298.
- [27] D. RIDZAL, *Trust-Region SQP Methods with Inexact Linear System Solves for Large-Scale Optimization*, PhD thesis, Rice University, 2006.
- [28] P. SARMA, L. DURLOFSKY, AND K. AZIZ, *Kernel principal component analysis for efficient, differentiable parameterization of multipoint geostatistics*, Mathematical Geosciences, 40 (2008), pp. 3–32. 10.1007/s11004-007-9131-7.
- [29] O. SCHENK AND K. GÄRTNER, *On fast factorization pivoting methods for sparse symmetric indefinite systems*, Elec. Trans. Numer. Anal., 23 (2006), pp. 158–179.
- [30] R.A. SCHOWENGERDT, *Remote Sensing: Models and Methods for Image Processing*, Academic Press, 2006.
- [31] L. SIRGUE AND R.G. PRATT, *Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies*, Geophysics, 69 (2004), pp. 231–248.
- [32] M. TINKHAM, *Introduction to Superconductivity: Second Edition*, Dover books on physics and chemistry, Dover Publications, 2004.
- [33] A. WÄCHTER AND L. T. BIEGLER, *On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming*, Math. Program., 106 (2006), pp. 25–57.
- [34] S. XU AND G. LAMBARE, *Fast migration/inversion with multivalued rayfields. Part I: Method, validation test, and application in 2D to Marmousi*, Geophysics, 69 (2004), pp. 1311–1319.

LATEST PREPRINTS

No.	Author: Title
2013-01	H. Harbrecht, M. Peters <i>Comparison of Fast Boundary Element Methods on Parametric Surfaces</i>
2013-02	V. Bosser, A. Surroca <i>Elliptic logarithms, diophantine approximation and the Birch and Swinnerton-Dyer conjecture</i>
2013-03	A. Surroca Ortiz <i>Unpublished Talk: On some conjectures on the Mordell-Weil and the Tate-Shafarevich groups of an abelian variety</i>
2013-04	V. Bosser, A. Surroca <i>Upper bound for the height of S-integral points on elliptic curves</i>
2013-05	Jérémy Blanc, Jean-Philippe Furter, Pierre-Marie Poloni <i>Extension of Automorphisms of Rational Smooth Affine Curves</i>
2013-06	Rupert L. Frank, Enno Lenzmann, Luis Silvestre <i>Uniqueness of Radial Solutions for the Fractional Laplacian</i>
2013-07	Michael Griebel, Helmut Harbrecht <i>On the convergence of the combination technique</i>
2013-08	Gianluca Crippa, Carlotta Donadello, Laura V. Spinolo <i>Initial-Boundary Value Problems for Continuity Equations with BV Coefficients</i>
2013-09	Gianluca Crippa, Carlotta Donadello, Laura V. Spinolo <i>A Note on the Initial-Boundary Value Problem for Continuity Equations with Rough Coefficients</i>
2013-10	Gianluca Crippa <i>Ordinary Differential Equations and Singular Integrals</i>
2013-11	G. Crippa, M. C. Lopes Filho, E. Miot, H. J. Nussenzveig Lopes <i>Flows of Vector Fields with Point Singularities and the Vortex-Wave System</i>
2013-12	L. Graff, J. Fender, H. Harbrecht, M. Zimmermann <i>Key Parameters in High-Dimensional Systems with Uncertainty</i>
2013-13	Jérémy Blanc, Immanuel Stampfli <i>Automorphisms of the Plane Preserving a Curve</i>

LATEST PREPRINTS

- No.** **Author:** *Title*
- 2013-14 **Jérémy Blanc, Jung Kyu Canci**
Moduli Spaces of Quadratic Rational Maps with a Marked Periodic Point of Small Order
- 2013-15 **Marcus J. Grote, Johannes Huber, Drosos Kourounis, Olaf Schenk**
Inexact Interior-Point Method for Pde-Constrained Nonlinear Optimization