Finite Elements in Analysis and Design 228 (2024) 104052

Contents lists available at ScienceDirect

Finite Elements in Analysis & Design

journal homepage: www.elsevier.com/locate/finel

An inexact semismooth Newton method with application to adaptive randomized sketching for dynamic optimization ☆

Mohammed Alshehri [a,1], Harbir Antil [a,*,1], Evelyn Herberg [b,1], Drew P. Kouri [c,1]

[a] The Center for Mathematics and Artificial Intelligence (CMAI) and Department of Mathematical Sciences, George Mason University, Fairfax, VA 22030, USA
[b] The Interdisciplinary Center for Scientific Computing (IWR) Ruprecht-Karls-University Heidelberg, 69120 Heidelberg, Germany
[c] Sandia National Laboratories, P.O. Box 5800, MS 1320, Albuquerque, NM, 87185-1320, USA

ARTICLE INFO

MSC:
49L20
49M15
65K10
68W20
90C39
93C20

Keywords:
Nonsmooth optimization
Inexact gradient and Hessian
Semismooth Newton
Adaptivity
Convergence analysis
Compression methods
Randomized sketching
Measure control
Variational discretization

ABSTRACT

In many applications, one can only access the inexact gradients and inexact hessian times vector products. Thus it is essential to consider algorithms that can handle such inexact quantities with a guaranteed convergence to solution. An inexact adaptive and provably convergent semismooth Newton method is considered to solve constrained optimization problems. In particular, dynamic optimization problems, which are known to be highly expensive, are the focus. A memory efficient semismooth Newton algorithm is introduced for these problems. The source of efficiency and inexactness is the randomized matrix sketching. Applications to optimization problems constrained by partial differential equations are also considered.

1. Introduction

We introduce a memory-efficient sketched semismooth Newton method for solving dynamic optimization problems with form

$$\min_{u_i\in\mathbb{R}^{n_s},z_i\in\mathbb{R}^m}\sum_{i=1}^{n_t} f_i(u_{i-1},u_i,z_i) \quad \text{subject to} \quad c_i(u_{i-1},u_i,z_i)=0 \quad \text{and} \quad z_i\in\mathscr{Z}_{\text{ad},i}\subset\mathbb{R}^m, \tag{1.1}$$

☆ MA, HA, and EH's work was partially supported by National Science Foundation (NSF), USA grants DMS-2110263, DMS-1913004, the Air Force Office of Scientific Research (AFOSR), USA under Award NO: FA9550-22-1-0248, and Department of Navy, Naval PostGraduate School, USA under Award NO: N00244-20-1-0005. DPK's work was partially supported by AFOSR, USA and the Department of Energy, USA Office of Science Early Career Research Program.

* Corresponding author.
E-mail addresses: malsheh4@gmu.edu (M. Alshehri), hantil@gmu.edu (H. Antil), evelyn.herberg@iwr.uni-heidelberg.de (E. Herberg), dpkouri@sandia.gov (D.P. Kouri).

[1] The author names are arranged in an alphabetical order which is a standard practice in Mathematics.

https://doi.org/10.1016/j.finel.2023.104052
Received 14 July 2023; Received in revised form 21 September 2023; Accepted 22 September 2023
Available online 18 October 2023

where $u_i \in \mathbb{R}^{n_s}$ is the state variable, $z_i \in \mathbb{R}^m$ is the control variable and $\mathscr{Z}_{\mathrm{ad},i} \subset \mathbb{R}^m$ is the admissible control set at the ith time step, $i = 1, \dots, n_t$. We assume that $\mathscr{Z}_{\mathrm{ad},i}$ has the following structure

$$\mathscr{Z}_{\mathrm{ad},i} := \left\{ z_i \in \mathbb{R}^m \mid h_i(z_i) = 0 \quad \text{and} \quad g_i(z_i) \le 0 \right\}, \tag{1.2}$$

where $h_i : \mathbb{R}^m \to \mathbb{R}^{n_{\mathrm{eq},i}}$ and $g_i : \mathbb{R}^m \to \mathbb{R}^{n_{\mathrm{ineq},i}}$. Furthermore, $u_0 \in \mathbb{R}^{n_s}$ is the given initial state, $f_i : \mathbb{R}^{n_s \times n_s \times n_t} \to \mathbb{R}$ denotes the objective function associated with ith control and state and $c_i : \mathbb{R}^{n_s \times n_s \times m} \to \mathbb{R}^{n_s}$ is the time-discrete dynamical system, which advances the state from u_{i-1} to u_i. Dynamic optimization problems like (1.1) arise in many applications, including, the control of pathogen propagation in built environment [1,2], energy system operations [3], vortex control in nuclear reactors and superconductors [4], magnetic drug targeting [5], and full waveform inversion [6].

These problems suffer from many computational challenges. Gradient-based methods for solving problems of the form (1.1) require storing the entire state trajectory $\{u_1, \dots, u_{n_t}\}$ and auxiliary variables such as Lagrange multipliers, incurring a storage cost of $\mathcal{O}(n_t(2n_s + m))$. When the system $c_i(u_{i-1}, u_i, z_i) = 0$ is uniquely solvable, we can write the minimization problem (1.1) in terms of controls $\{z_1, \dots, z_{n_t}\}$, leading to the so-called reduced formulation. In this setting, the evaluation of the gradient requires the solution of the adjoint or costate equation backward in time, which in turn depends on the state trajectory u_i. A naïve implementation of this again incurs the storage cost of $\mathcal{O}(n_t(n_s + m))$, which can be prohibitively expensive. Consider the discretized pathogen propagation problem described in [1], where $n_s \approx 10^9$ and $n_t \approx 10^6$. Using such an implementation, a gradient evaluation for this application would require the storage of roughly 10^{15} floating point numbers.

One can reduce the storage cost, for instance, using reduced-order models (ROMs) [7–12]. However, the approximation capabilities of a fixed ROM can significantly degrade as the optimization progresses. To overcome this, [13,14] employ trust-region methods to adaptively control the fidelity of the ROM. Nevertheless, these approaches tend to be intrusive and do not easily adapt to legacy codes. Indeed, almost all existing ROM approaches require extra reduction steps to approximate nonlinear terms in PDEs [15–17]. For dynamic optimization problems, one can also use checkpointing to reduce the memory requirements of the state trajectory [18–20]. However, checkpointing may increase the gradient computation costs as discussed in these references.

The contributions in this article are twofold. First, we introduce an inexact adaptive semismooth Newton method and provide a convergence analysis of this algorithm for generic optimization problems in function spaces. Secondly, we consider dynamic optimization problems of the form (1.1), where inexactness arises due to matrix sketching. To reduce the memory requirements for (1.1), we use adaptive randomized sketching to compress the state $\{u_1, \dots, u_{n_s}\}$ as in [21]. Matrix sketching leads to low-rank approximations of the state [22–24]. In our setting, this further leads to inexact gradients and Hessian applications. The latter approximation are carefully controlled within the adaptive semismooth Newton method. This results in a provably convergent and efficient algorithm for (1.1). We refer to [21] for the initial work on adaptive sketching trust-region methods for solving unconstrained dynamic optimization problems.

The remainder of the paper is organized as follows. In Section 2, we describe the finite-dimensional optimization problem (1.1), state its optimality conditions, and review randomized matrix sketching. In Section 3, we introduce an inexact semismooth Newton method with guaranteed superlinear convergence in Banach space. Our convergence proof is adapted from [25, Sect. 3.2.4]. In Section 4, we apply our semismooth Newton method to a generic optimal control problem with inexact gradient and Hessian information. This problem is motivated by sketching for (1.1). To this end, we derive approximation estimates for the gradient and Hessian computations. Although some of these results are known in the literature, we provide complete details, including proofs, whenever appropriate, in Appendix A. The general nature of these results enable use to apply the semismooth Newton method to our target dynamic optimization problem (1.1). Finally, we demonstrate the numerical performance of our algorithm on an initial measure control problem in Section 5.

2. Problem formulation and randomized sketching

Throughout, we consider the reduced form of (1.1). To this end, we assume that the equality constraint $c_i(u_{i-1}, u_i, z_i) = 0$ in (1.1) is uniquely solvable for fixed u_{i-1} and z_i. Although, one can solve (1.1) using sequential quadratic programming methods, the memory burden of $\mathcal{O}(n_t(2n_s + m))$ floating point numbers renders such methods infeasible for many practical applications. To describe the reduced formulation, we collect the controls and states into stacked column vectors, denoted by

$$\mathbf{u} = [u_1^\top, \dots, u_{n_t}^\top]^\top \in \mathscr{U} := \mathbb{R}^{n_s n_t} \qquad \text{and} \qquad \mathbf{z} = [z_1^\top, \dots, z_{n_t}^\top]^\top \in \mathscr{Z} := \mathbb{R}^{m n_t},$$

and represent the objective function, state equation and constraint functions as

$$f(\mathbf{u}, \mathbf{z}) := \sum_{i=1}^{n_t} f_i(u_i, z_i), \qquad \mathbf{c}(\mathbf{u}, \mathbf{z}) := \begin{bmatrix} c_1(u_0, u_1, z_1) \\ \vdots \\ c_{n_t}(u_{n_t-1}, u_{n_t}, z_{n_t}) \end{bmatrix} = 0,$$

$$\mathbf{h}(\mathbf{z}) := \begin{bmatrix} h_1(z_1) \\ \vdots \\ h_{n_t}(z_{n_t}) \end{bmatrix} = 0, \qquad \text{and} \qquad \mathbf{g}(\mathbf{z}) := \begin{bmatrix} g_1(z_1) \\ \vdots \\ g_{n_t}(z_{n_t}) \end{bmatrix} \le 0.$$

Employing this notation, we can rewrite (1.1) as

$$\min_{(\mathbf{u},\mathbf{z}) \in \mathscr{U} \times \mathscr{Z}} f(\mathbf{u}, \mathbf{z}) \qquad \text{subject to} \qquad \mathbf{c}(\mathbf{u}, \mathbf{z}) = 0 \quad \text{and} \quad \mathbf{z} \in \mathscr{Z}_{\mathrm{ad}}, \tag{2.1}$$

where

$$\mathscr{Z}_{\text{ad}} := \mathscr{Z}_{\text{ad},1} \times \cdots \times \mathscr{Z}_{\text{ad},n_t} = \{\mathbf{z} \in \mathscr{Z} \mid \mathbf{h}(\mathbf{z}) = 0 \quad \text{and} \quad \mathbf{g}(\mathbf{z}) \leq 0\}$$

is the admissible control set.

To generate the reduced problem, we assume that f and $\mathbf{c}$ are continuously differentiable on $\mathscr{U} \times \mathscr{Z}$ and that there exists a unique control-to-state map $\mathbf{z} \mapsto S(\mathbf{z}) : \mathscr{Z} \to \mathscr{U}$. Here, $S(\mathbf{z})$ is the unique state trajectory satisfying $\mathbf{c}(S(\mathbf{z}), \mathbf{z}) = 0$ holds for each $\mathbf{z} \in \mathscr{Z}$. To this end, we assume that the state Jacobian of the state equation function $\mathbf{c}$, denoted by $d_{\mathbf{u}}\mathbf{c}(S(\mathbf{z}), \mathbf{z})$, has a bounded inverse for all controls $\mathbf{z} \in \mathscr{Z}$. The unique state trajectory takes the form

$$S(\mathbf{z}) := \begin{bmatrix} S_1(u_0, z_1) \\ S_2(S_1(u_0, z_1), z_2) \\ \vdots \\ S_{n_t}(S_{n_t-1}(\dots, z_{n_t-1}), z_{n_t}) \end{bmatrix}.$$

By the implicit function theorem [26, Th. 1.41], we have that all S_i and S are continuously differentiable. Consequently, we can formulate the reduced problem

$$\min_{\mathbf{z} \in \mathscr{Z}_{\text{ad}}} \{\hat{f}(\mathbf{z}) := f(S(\mathbf{z}), \mathbf{z})\}. \tag{2.2}$$

To derive the optimality conditions for (2.2), we define the Lagrangian functional

$$\mathscr{L}(\mathbf{z}, \lambda, \boldsymbol{\mu}) := \hat{f}(\mathbf{z}) + \lambda^{\top}\mathbf{h}(\mathbf{z}) + \boldsymbol{\mu}^{\top}\mathbf{g}(\mathbf{z}),$$

where $\lambda \in \mathbb{R}^{n_{\text{eq}}}$ and $\boldsymbol{\mu} \in \mathbb{R}^{n_{\text{eq}}}$ are Lagrange multipliers. Here, $n_{\text{eq}} := n_{\text{eq},1} + \cdots + n_{\text{eq},n_t}$ and $n_{\text{ineq}} := n_{\text{ineq},1} + \cdots + n_{\text{ineq},n_t}$. Assuming that $\hat{f}$, $\mathbf{h}$ and $\mathbf{g}$ are twice continuously differentiable, we deduce the optimality (KKT) conditions

$$\nabla_{\mathbf{z}}\mathscr{L}(\mathbf{z}, \lambda, \boldsymbol{\mu}) = \nabla \hat{f}(\mathbf{z}) + \nabla \mathbf{h}(\mathbf{z})\lambda + \nabla \mathbf{g}(\mathbf{z})\boldsymbol{\mu} = 0, \tag{2.3a}$$

$$\mathbf{h}(\mathbf{z}) = 0, \tag{2.3b}$$

$$\boldsymbol{\mu} \geq 0, \quad \mathbf{g}(\mathbf{z}) \leq 0, \quad \boldsymbol{\mu}^{\top}\mathbf{g}(\mathbf{z}) = 0. \tag{2.3c}$$

We equivalently reformulate the complementary conditions (2.3c) as

$$\mathbf{F}_3(\mathbf{z}, \boldsymbol{\mu}) := \max\{0, \boldsymbol{\mu} + \kappa \mathbf{g}(\mathbf{z})\} - \boldsymbol{\mu} = 0,$$

for fixed $\kappa > 0$. Recall that $\mathbf{F}_3$ is semismooth [25, Def. 2.5], since the maximum function is semismooth and g is continuously differentiable. We can then rewrite the optimality conditions in (2.3) as the semismooth nonlinear system of equations

$$\mathbf{F}(\mathbf{z}, \lambda, \boldsymbol{\mu}) := \begin{bmatrix} \nabla_{\mathbf{z}}\mathscr{L}(\mathbf{z}, \lambda, \boldsymbol{\mu}) \\ h(\mathbf{z}) \\ \mathbf{F}_3(\mathbf{z}, \boldsymbol{\mu}) \end{bmatrix} = 0. \tag{2.4}$$

We will solve (2.4) using the semismooth Newton's method.

As seen in the definition of $\hat{f}$, we can evaluate the objective function sequentially while only storing two state variables u_{i-1} and u_i. Recall that u_{i-1} is needed to compute u_i by solving the ith equality constraint. Unfortunately, the gradient of $\hat{f}$ in (2.4) requires the solution of the adjoint equation

$$d_{\mathbf{u}}\mathbf{c}(S(\mathbf{z}), \mathbf{z})^{*}\mathbf{p} = -d_{\mathbf{u}}f(S(\mathbf{z}), \mathbf{z}), \tag{2.5}$$

or equivalently

$$(d_{u_i}c_i(u_{i-1}, u_i, z_i))^{*}p_i = -d_{u_i}f_i(u_i, z_i) - (d_{u_i}c_{i+1}(u_i, u_{i+1}, z_i))$$

for $i = n_t - 1, \dots, 1$, which is solved backward in time. Upon first glance, we require the entire state trajectory to compute the adjoint trajectory. As mentioned earlier, this storage requirement can be alleviated using checkpointing, resulting in additional computational cost. Instead, we use randomized matrix sketching to compress the state trajectory $\mathbf{u} = S(\mathbf{z})$.

2.1. Randomized matrix sketching

Let $\mathbf{U} = [\mathbf{u}_1 \cdots \mathbf{u}_{n_t}] \in \mathbb{R}^{n_s \times n_t}$ be the state trajectory $\mathbf{u}$ reshaped as a matrix. We denote the prescribed sketch rank by $r \in \mathbb{N}$ and assume that the sketching parameters $k, s \in \mathbb{N}$ satisfy

$$r \leq k \leq s \leq \min\{M, N\}.$$

In our numerical experiments, we set $k = 2r + 1$ and $s = 2k + 1$. We fix four random linear dimension-reduction maps with i.i.d. standard normal entires:

$$\boldsymbol{\Upsilon} \in \mathbb{R}^{k \times n_s}, \quad \boldsymbol{\Omega} \in \mathbb{R}^{k \times n_s}, \quad \boldsymbol{\Phi} \in \mathbb{R}^{s \times n_t}, \quad \boldsymbol{\Psi} \in \mathbb{R}^{s \times n_t}.$$

The rank-r sketch of the matrix $\mathbf{U}$, denoted $\{\{\mathbf{U}\}\}_r$, is the triplet of matrices $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$, given by

$$\begin{aligned}
\mathbf{X} &= \boldsymbol{\Upsilon}\mathbf{U} \in \mathbb{R}^{k\times n_t} && \text{(co-range sketch)},\\
\mathbf{Y} &= \mathbf{U}\boldsymbol{\Omega}^* \in \mathbb{R}^{n_s\times k} && \text{(range sketch)},\\
\mathbf{Z} &= \boldsymbol{\Phi}\mathbf{U}\boldsymbol{\Psi}^* \in \mathbb{R}^{s\times s} && \text{(core sketch)}.
\end{aligned}$$

These computations can be performed online. For example, with the representation $\mathbf{U} = \begin{bmatrix}\mathbf{u}_1 \dots \mathbf{u}_{n_t}\end{bmatrix}$, we can calculate

$$\mathbf{X}^{(0)} = \mathbf{0}, \qquad \mathbf{X}^{(i)} = \mathbf{X}^{(i-1)} + \boldsymbol{\Upsilon}\mathbf{u}_i\mathbf{e}_i^\top \qquad i = 1, \dots, n_t,$$

where $\mathbf{e}_i$ denotes the ith unit vector. Similar recursions hold for computing $\mathbf{Y}$ and $\mathbf{Z}$. Therefore, we do not need to store $\mathbf{U}$ to build the sketch $\{\{\mathbf{U}\}\}_r$, which is critical for practical applications.

To recover the sketched columns of $\mathbf{U}$, we first compute the QR decompositions

$$\begin{aligned}
\mathbf{X}^* &=: \mathbf{P}\mathbf{R}_1, \quad \text{where} \quad \mathbf{P} \in \mathbb{R}^{n_t\times k},\\
\mathbf{Y} &=: \mathbf{Q}\mathbf{R}_2, \quad \text{where} \quad \mathbf{Q} \in \mathbb{R}^{n_s\times k},
\end{aligned}$$

and use the core sketch $\mathbf{Z}$ to compute a core approximation by solving two small least-squares problems

$$\mathbf{C} := (\boldsymbol{\Phi}\mathbf{Q})^\dagger \mathbf{Z}((\boldsymbol{\Psi}\mathbf{P})^\dagger)^* \in \mathbb{R}^{k\times k}.$$

The rank-k approximation of $\mathbf{U}$ is then given by

$$\widetilde{\mathbf{U}} := \mathbf{Q}\mathbf{C}\mathbf{P}^*.$$

We define $\mathbf{W} := \mathbf{C}\mathbf{P}^* \in \mathbb{R}^{k\times n_t}$ and store only the skinny matrices $\mathbf{Q}$ and $\mathbf{W}$, which require $\mathcal{O}(k(n_s+n_t))$ memory, instead of $\mathcal{O}(n_s n_t)$ for the storage of $\mathbf{U}$. However, since $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ need to be stored intermediately during the column wise updating process, the required storage is marginally increased to $\mathcal{O}(k(n_s+n_t)+s^2)$. The reconstruction of $\widetilde{\mathbf{U}}$ from $\mathbf{Q}$ and $\mathbf{W}$ can then be performed column-wise as $\widetilde{\mathbf{U}}[:, j] = \mathbf{Q}\mathbf{W}[:, j]$.

Before concluding this discussion, we recall that the sketching error (cf. [21, Th. 3.4]) satisfies

$$\mathbb{E}_{\boldsymbol{\Upsilon},\boldsymbol{\Omega},\boldsymbol{\Phi},\boldsymbol{\Psi}}\left[\|\mathbf{U} - \widetilde{\mathbf{U}}\|_F\right] \le \sqrt{6}\cdot\tau_{r+1}(\mathbf{U}), \tag{2.6}$$

where $\tau_{r+1}(\mathbf{U})$ denotes the $(r+1)$-st tail energy:

$$\tau_{r+1}(\mathbf{U}) := \min_{\operatorname{rank}(\mathbf{B})<r+1} \|\mathbf{U}-\mathbf{B}\|_F = \left(\sum_{i\ge r+1}\sigma_i^2(\mathbf{U})\right)^{1/2},$$

with σ_i denoting the ith singular value. In particular, the rank-k sketching approximation $\widetilde{\mathbf{U}}$ differs from the best rank-r approximation by a constant factor on average.

2.2. Approximation of gradient and Hessian for (2.2)

Let $\widetilde{\mathbf{u}}$ denotes the approximation of the state due to sketching and let the corresponding approximate adjoint be given by $\widetilde{\mathbf{p}}$, i.e., $\widetilde{\mathbf{p}}$ solves (2.5) with $\mathbf{u}$ replaced by $\widehat{\mathbf{u}}$. While solving the state equation, we can calculate the objective function f as in [21, Alg. 4.1]. However, due to sketching, the adjoint $\widetilde{\mathbf{p}}$ is inexact and subsequently the gradient $\nabla\hat{f}(\mathbf{z})$ and Hessian $\nabla^2\hat{f}(\mathbf{z})$ are inexact. Recall that applying the Hessian to a vector generally depends on the entire state and adjoint trajectories as well as trajectories from two additional dynamical systems. Consequently, one must sketch two additional trajectories to apply the Hessian, cf. [21, Alg. A.6].

In order to prove convergence of our semismooth Newton method, we must quantify and control the sketching errors in the gradient and Hessian computations. To this end, we denote the approximate gradient by $\mathbf{g}_r(\mathbf{z})$ and the approximate Hessian by $\mathbf{H}_r(\mathbf{z})$. Let $\widetilde{\mathbf{F}}(\mathbf{z},\lambda,\boldsymbol{\mu})$ denote $\mathbf{F}(\mathbf{z},\lambda,\boldsymbol{\mu})$ in (2.4) with $\nabla\hat{f}(\mathbf{z})$ replaced with $\mathbf{g}_r(\mathbf{z})$ in the first equation. Then it is straightforward to see that

$$\widetilde{\mathbf{F}}(\mathbf{z},\lambda,\boldsymbol{\mu}) - \mathbf{F}(\mathbf{z},\lambda,\boldsymbol{\mu}) = \begin{pmatrix}\mathbf{g}_r(\mathbf{z}) - \nabla\hat{f}(\mathbf{z})\\ 0\\ 0\end{pmatrix}$$

and setting $\epsilon = \widetilde{\mathbf{F}}(\mathbf{z},\lambda,\boldsymbol{\mu}) - \mathbf{F}(\mathbf{z},\lambda,\boldsymbol{\mu})$, we obtain

$$\|\epsilon\|_2 = \|\mathbf{g}_r(\mathbf{z}) - \nabla\hat{f}(\mathbf{z})\|_2. \tag{2.7}$$

Furthermore, let $M \in \partial^{cl}\mathbf{F}(\mathbf{z},\lambda,\boldsymbol{\mu})$, where $\partial^{cl}\mathbf{F}$ denotes the set-valued map of Clarke's generalized Jacobians of $\mathbf{F}$ [25, Def. 2.1], i.e.,

$$\partial^{cl}\mathbf{F}(\mathbf{z}) := \operatorname{conv}\Big\{\ M \in \mathbb{R}^{mn_t\times mn_t} \mid \mathbf{F} \text{ differentiable at } \mathbf{z}^k \text{ for all } k \text{ with } \mathbf{z}^k \overset{k\to\infty}{\longrightarrow} \mathbf{z}, \tag{2.8}$$

$$\text{and } \mathbf{F}'(\mathbf{z}^k) \overset{k\to\infty}{\longrightarrow} M\ \Big\}. \tag{2.9}$$

Here, conv denotes the convex hull. Since $\hat{f}$, $\mathbf{h}$ and $\mathbf{g}$ are assumed to be twice continuously differentiable, M has the form

$$M = \begin{bmatrix} \nabla^2_{\mathbf{z}} \mathscr{L}(\mathbf{z}, \lambda, \boldsymbol{\mu}) & \nabla \mathbf{h}(\mathbf{z}) & \nabla \mathbf{g}(\mathbf{z}) \\ \nabla \mathbf{h}(\mathbf{z})^\top & 0 & 0 \\ M_{\mathbf{z}} & 0 & M_{\boldsymbol{\mu}} \end{bmatrix},$$

where $M_{\mathbf{z}}$ denotes the Clarke generalized Jacobian of $\mathbf{F}_3$ with respect to $\mathbf{z}$ and $M_{\boldsymbol{\mu}}$ denotes the Clarke generalized Jacobian of $\mathbf{F}_3$ with respect to $\boldsymbol{\mu}$.

Let $\widetilde{M}$ denote the sketching approximation of M and note that the only error occurs is in $\nabla^2 \hat{f}(\mathbf{z})$ (i.e., the (1,1)-block of M) and so we obtain

$$\widetilde{M} - M = \begin{bmatrix} \mathbf{H}_r(\mathbf{z}) - \nabla^2 \hat{f}(\mathbf{z}) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Setting $\Delta = \widetilde{M} - M$, we see that

$$\|\Delta \mathbf{v}\|_2 = \|(\mathbf{H}_r(\mathbf{z}) - \nabla^2 \hat{f}(\mathbf{z}))v_1\|_2 \tag{2.10}$$

for any $\mathbf{v} = [v_1^\top, v_2^\top, v_3^\top]^\top \in \mathbb{R}^{mn_t + n_{\text{eq}} + n_{\text{ineq}}}$. From (2.7) and (2.10), we notice that the approximation errors in the value and generalized Jacobians of $\mathbf{F}$ in (2.4) only depend on the gradient and Hessian of $\hat{f}(\mathbf{z})$. In Section 4, we will derive error estimates for these components in order to guarantee convergence of our inexact semismooth Newton method.

The formulation discussed in this section is for finite-dimensional optimization problems, which are a special case of the infinite-dimensional setting discussed in the next section. One advantage of the infinite-dimensional setting is that it enables the treatment of inexactness arising from sources other than just matrix sketching.

3. Abstract semismooth Newton with inexact values and Jacobians

Let $(X, \|\cdot\|_Z)$ and $(Y, \|\cdot\|_Y)$ be real Banach spaces. We will develop an inexact semismooth Newton algorithm for solving the abstract problem of finding $x \in X$ such that

$$F(x) = 0, \tag{3.1}$$

where $F : X \to Y$ is a ∂^*-semismooth map. Here, ∂^* denotes an appropriate generalized derivative operator. For example, in finite dimensions, one can take $\partial^* = \partial^{cl}$ as done in Section 2. See [25, Def. 3.1] for more information on ∂^*-semismoothness.

As seen in Section 2, the problem (2.1) can be formulated as a special case of (3.1). More generally, we can formulate a certain class of simulation-constrained optimization problems in an analogous way. Let $(U, \|\cdot\|_Y)$, $(W, \|\cdot\|_W)$, $(Z, \|\cdot\|_Z)$ and $(H, \|\cdot\|_H)$ be real Banach spaces and let $(G, \|\cdot\|_G)$ be a real Hilbert space. We denote the topological dual spaces associated with H, G, U, W, and Z by H^*, G^*, U^*, W^* and Z^*, respectively. To generalize (2.1), we denote the optimization variables by u (state) and z (control/design), and consider the problem

$$\min_{(u,z)\in U\times Z} J(u, z) \qquad \text{subject to} \qquad e(u, z) = 0 \qquad \text{and} \qquad z \in Z_{\text{ad}}, \tag{$\mathcal{P}$}$$

where $J : U \times Z \to \mathbb{R}$ denotes the cost functional, $e : U \times Z \to W$ is the state equation, $h : Z \to H$ is an auxiliary equality constraint, $g : Z \to G$ is an inequality constraint and the admissible control set is

$$Z_{\text{ad}} = \{z \in Z \mid h(z) = 0 \quad \text{and} \quad g(z) \in K\}. \tag{3.2}$$

Here, $K \subset G$ is a nonempty, closed and convex cone. As before, we assume that the state equation $e(u, z) = 0$ has a unique solution $u(z) \in U$ for every control $z \in Z_{\text{ad}}$, which enables us to define the reduced objective function $\hat{J}(z) := J(u(z), z)$ and the reduced optimization problem

$$\min_{z\in Z_{\text{ad}}} \hat{J}(z). \tag{$\hat{\mathcal{P}}$}$$

To construct F for this application, we introduce the Lagrangian functional

$$\mathscr{L}(z, \lambda, \mu) := \hat{J}(z) + \langle h(z), \lambda\rangle_{H,H^*} + \langle g(z), \mu\rangle_{G,G^*}$$

for multipliers $\lambda \in H^*$ and $\mu \in G^*$. Since G is a Hilbert space, we identify its dual G^* with G and thus replace the duality pairing $\langle g(z), \mu\rangle_{G,G^*}$ by scalar product $(g(z), \mu)_G$.

Assuming $\hat{J}$, h, and g are continuously differentiable, the optimality conditions for ($\mathcal{P}$) are given by

$$\nabla_z \mathscr{L}(z, \lambda, \mu) = \hat{J}'(z) + h'(z)^*\lambda + g'(z)^*\mu = 0, \tag{3.3a}$$

$$h(z) = 0, \tag{3.3b}$$

$$g(z) \in K, \quad \mu \in K^-, \quad \text{and} \quad (g(z), \mu)_G = 0, \tag{3.3c}$$

where $K^- := \{x \in G \mid (x, k)_G \le 0, \ \forall k \in K\}$ denotes the *polar cone*. As in (2.3), we can write the complementarity condition in (3.3c) in an equivalent form, which we state in Theorem 3.1. The proof of this result has been provided in Appendix B.

Theorem 3.1. *The complementarity conditions in (3.3c) are equivalent to*

$$F_3(z,\mu) := \mathrm{proj}_{K^-}(\mu + \kappa g(z)) - \mu = 0\,, \tag{3.4}$$

for fixed $\kappa > 0$, where $\mathrm{proj}_{K^-}(x)$ denotes the metric projection of $x \in G$ onto the polar cone K^-.

In view of Theorem 3.1, the optimality conditions in (3.3) are equivalent to

$$F(x) = \begin{bmatrix} \mathscr{L}_z(z,\lambda,\mu) \\ h(z) \\ F_3(z,\mu) \end{bmatrix} = 0\,. \tag{3.5}$$

Provided that the projection onto the polar cone K^- is semismooth, which is the case in (2.4) (i.e., $K = (-\infty, 0]^{n_{\mathrm{ineq}}}$ and $K^- = [0,\infty)^{n_{\mathrm{ineq}}}$), and $\widehat{J}$, h, and g are twice continuously differentiable, we can apply a semismooth Newton method to solve (3.5).

Motivated by the inexactness introduced by sketching, we are interested in rigorously handling inexactness in the evaluation of F and its generalized Jacobians. Returning to the general setting of (3.1), we focus on $F : X \to Y$ that is not differentiable in the classical sense but is rather ∂^*-semismooth. Let $V \subset X$ be an open set and consider a set-valued map $\partial^* F : V \rightrightarrows \mathcal{L}(X,Y)$ with nonempty images $\partial^* F(x) \neq \emptyset$ for all $x \in V$. Here, $\partial^* F(x)$ consists of generalized Jacobians of F at x and $L(X,Y)$ denotes the space of bounded linear operators that map X into Y. We say that F is $\partial^* F$-semismooth (cf. [25, Def. 3.1]) at $x \in V$ if F is continuous near x and

$$\sup_{M \in \partial^* F(x+s)} \|F(x+s) - F(x) - Ms\|_Y = o(\|s\|_X) \quad \text{as} \quad \|s\|_X \to 0.$$

The kth semismooth Newton step s_k for (3.1) is obtained by solving

$$M^k s^k = -F(x^k),$$

where $M^k \in \partial^* F(x^k)$. Now, for all k, let

$$\widetilde{M}^k := M^k + \Delta^k, \qquad \widetilde{F}_k(x^k) := F(x^k) + \epsilon^k,$$

where Δ^k and ϵ^k denote the errors in the generalized Jacobian and the function value, respectively. Using this notation, we introduce the inexact semismooth Newton's method for solving (3.1) in Algorithm 1.

Algorithm 1 Inexact Semismooth Newton's Method

Require: The initial guess $x^0 \in X$

1: **for** $k = 0, 1, \ldots$ **do**
2: Choose $\widetilde{M}^k \in L(X,Y)$ boundedly invertible.
3: Obtain $s^k \in X$ by solving $\widetilde{M}^k s^k = -\widetilde{F}_k(x^k)$.
4: Set $x^{k+1} = x^k + s^k$.
5: If $x^{k+1} = x^k$, STOP with $x^\star = x^{k+1}$.
6: **end for**

Algorithm 1 is an inexact version of the traditional semismooth Newton method. Consequently, the convergence proof is closely related to known results in the literature, see for instance [25, Sec. 3.2.4]. We emphasize that both Newton and quasi-Newton methods (such as BFGS) are special cases of the semismooth Newton's method, hence a convergence result of the inexact semismooth Newton's method, Algorithm 2, can also be applied to show convergence of these methods.

A few comments are in order regarding Algorithm 1. First, the stopping criterion in Step 5 ensures local convergence. One could also consider a more traditional stopping criterion based on $\widetilde{F}(x^k) = 0$ and still establish convergence, see for instance [25, Rem. 3.11 and Th. 3.15]. Secondly, in [25, Sec. 3.2.4.], a smoothing step is inserted after Step 3 to weaken the invertibility condition and to account for appropriate function spaces in certain cases. For simplicity of presentation, we have omitted this step, but the entire discussion below works in this case after minor modification.

Our proof below has been adapted from [25, Sec. 3.2.4], which examines inexactness in the generalized derivative (Δ^k). Here, we additionally allow an error ϵ^k in the function F. This has further consequences as stated in Assumption 3.2(c) below.

Assumption 3.2. We require the following conditions to hold.

(a) For all k, the mapping $M^k \in L(X,Y)$ is continuous with bounded inverses, and there exists a constant $C_{M^{-1}} > 0$, independent of k, such that

$$\|(M^k)^{-1}\|_{L(X,Y)} \leq C_{M^{-1}}\,.$$

(b) There exists $M^k \in \partial^* F(x^k)$, which fulfills the requirements from (a), such that for $\Delta^k := \widetilde{M}^k - M^k$ we have

$$\|\Delta^k s^k\|_Y = o(\|s^k\|_X) \qquad \text{as} \qquad \|s^k\|_X \to 0.$$

(c) The residual approximation $\widetilde{F}_k(x^k)$ with $\epsilon^k := \widetilde{F}_k(x^k) - F(x^k)$ satisfies

$$\|\epsilon^k\|_Y = o(\|\widetilde{F}_k(x^k)\|_Y) \qquad \text{as} \qquad \|\widetilde{F}_k(x^k)\|_Y \to 0.$$

Remark 3.3. We note that condition (c) in Assumption 3.2 is stronger than what is required by inexact trust-region methods [27,28]. In contrast, trust-region methods enforce

$$\|\epsilon^k\|_Y = \mathcal{O}(\|\widetilde{F}_k(x^k)\|_Y).$$

On the other hand, condition (b) in Assumption 3.2 is closely related to the Dennis-Moré condition [29,30], which is required to obtain the superlinear convergence rate.

Under Assumption 3.2, we have the following convergence result; see also [25, Th. 3.18].

Theorem 3.4. *Let Assumption 3.2 hold, $F : V \to Y$, where $V \subset X$ is open, and let the generalized Jacobian set-valued map $\partial^* F : V \rightrightarrows L(X,Y)$ have nonempty images. Let $\bar{x} \in V$ solve (3.1) and F be Lipschitz continuous near $\bar{x}$. If F is $\partial^* F$-semismooth at $\bar{x}$, then there exists $\delta > 0$, sufficiently small, such that for all $x^0 \in \bar{X} + \delta B_X$, Algorithm 1 either:*

(i) *Terminates with $x^\star = x^k = \bar{x}$, or*
(ii) *Generates a sequence $\{x^k\} \subset V$ that converges q-superlinearly to $\bar{x}$ in X.*

Proof. The proof of Theorem 3.4 is related to the proof of [25, Th. 3.18]. The main difference is the introduction of the error term ϵ^k and the omission of the smoothing step. We provide details for completeness.

To prove this result, we first prove a bound on the norm of s^k. The triangle inequality, Assumption 3.2(a) and $\widetilde{M}^k s^k = -\widetilde{F}_k(x^k)$ ensure that

$$\begin{aligned}\|s^k\|_X &\le \|(M^k)^{-1}\|_{L(X,Y)} \left(\|\Delta^k s^k\|_Y + \|\widetilde{M}^k s^k\|_Y\right) \\ &\le C_{M^{-1}} \left(\|\Delta^k s^k\|_Y + \|\epsilon^k\|_Y + \|F(x^k)\|_Y\right). \end{aligned} \tag{3.6}$$

By Assumption 3.2(b), we can choose $\delta > 0$ sufficiently small so that

$$C_{M^{-1}} \|\Delta^k s^k\|_Y \le \frac{1}{2}\|s^k\|_X.$$

To bound the second term on the right-hand side of (3.6), Assumption 3.2(c) implies the existence of a nonnegative sequence $\{\eta^k\}$ with $\eta^k \to 0$ such that

$$\|\epsilon^k\|_Y \le \eta^k \|\widetilde{F}_k(x^k)\|_Y.$$

Consequently, for any fixed $\eta \in (0,1)$, there exists $K_\eta \in \mathbb{N}$ such that $\eta^k \le \eta$ for all $k \ge K_\eta$. This and Lipschitz continuity of F near $\bar{x}$ ensure that

$$\|\epsilon^k\|_Y \le \eta^k \|\widetilde{F}_k(x^k) - F(x^k)\|_Y + \eta^k \|F(x^k) - F(\bar{x})\|_Y \le \eta \|\epsilon^k\|_Y + \eta^k L \|x^k - \bar{x}\|_X$$

implying

$$\|\epsilon^k\|_Y \le \eta^k \frac{L}{1-\eta} \|x^k - \bar{x}\|_X,$$

where $L > 0$ is the Lipschitz modulus of F around $\bar{x}$, i.e.

$$\|F(x^k)\|_Y = \|F(x^k) - F(\bar{x})\|_Y \le L\|x^k - \bar{x}\|_X.$$

Here, we shrink δ as needed to ensure Lipschitz continuity. Combining these estimates, we have that $\|s^k\|_X$ satisfies

$$\|s^k\|_X \le 2LC_{M^{-1}} \left(\frac{\eta^k}{1-\eta} + 1\right) \|x^k - \bar{x}\|_X \le 2LC_{M^{-1}} \left(\frac{\eta}{1-\eta} + 1\right) \|x^k - \bar{x}\|_X \qquad \forall\, k \ge K_\eta.$$

Now, we bound the norm of $x^{k+1} - \bar{x}$. To this end, we have that

$$M^k(x^{k+1} - \bar{x}) = -\Delta^k s^k + M^k(x^k - \bar{x}) - \epsilon^k - F(x^k) + F(\bar{x}).$$

Therefore, we can bound the norm of $x^{k+1} - \bar{x}$ as

$$\|x^{k+1} - \bar{x}\|_X \le C_{M^{-1}} \left(\|\Delta^k s^k\|_Y + \|F(x^k) - F(\bar{x}) - M^k(x^k - \bar{x})\|_Y + \|\epsilon^k\|_Y\right).$$

By the preceding arguments for bounding $\|s^k\|_X$, we see that the first term is bounded as above and the third is $o(\|x^k - \bar{x}\|_X)$. In addition, the second term in the above bound is $o(\|x^k - \bar{x}\|_X)$ since F is $\partial^* F$-semismooth at $\bar{x}$. The desired result then follows as in the proof of Theorem 3.13 (a) of [25]. □

Remark 3.5. In practice, Algorithm 1 requires an extra step to determine a stepsize t^k so that $x^{k+1} = x^k + t^k s^k$ using, e.g., Armijo's rule [31, Eq. (3.4)]. As long as the stepsize t^k is admissible (to avoid steps that are too small), see, e.g., [32, Sec. 8.2], and $t^k \le 1$, the convergence result still holds true.

4. Application of Algorithm 1 to Simulation-Constrained Optimization

In this section, we consider (3.5) where approximations to F and its generalized Jacobians arise from approximating the state, adjoint and the other variables that arise while applying the Hessian of $\widehat{J}(z)$ to a vector. In this setting, $x = (z, \lambda, \mu)$, $X = Z \times H^* \times G$ and $Y = Z^* \times H \times G$. As in Section 2.2, we approximate F by replacing the state u with an approximation $\widetilde{u}$ in the derivative $\widehat{J}'(z)$, which we denote by $\widehat{J}'_r(z)$. Similarly, we approximate the generalized Jacobians of F by replacing the adjoint and other auxiliary variables in the application of the Hessian $\nabla^2\widehat{J}(z)$, which we denote by $\widehat{J}''_r(z)$. Consequently, the error committed by $\widetilde{F}(x)$ is equal to the error in the derivative approximation $\widehat{J}'_r(z)$ (cf. (2.7)) and similarly the error in the generalized Jacobian is equal to the error in the Hessian approximation $\widehat{J}''_r(z)$.

4.1. Approximation errors

We begin by deriving the expression of $\widehat{J}$ using the adjoint approach [26, Sec. 1.6.2], i.e.,

$$\widehat{J}'(z) = e_z(u(z), z)^* p(z) + J_z(u(z), z) \in Z^*, \tag{4.1a}$$

where $p(z) \in W^*$ solves the adjoint equation

$$0 = e_u(u(z), z)^* p(z) + J_u(u(z), z). \tag{4.1b}$$

We consider approximations of $\widehat{J}'(z)$ arising when we replace the state $u(z)$ in (4.1) with an approximation $\widetilde{u}$. We denote the resulting inexact derivative by $\widehat{J}'_r(z)$. In the context of sketching, $\widetilde{u}$ is a low-rank approximation of $u(z)$, cf. [21, Alg. 4.2].

As seen in [21, Sec. 4.3], we can obtain residual-based error indicators for $\widehat{J}'_r(z)$ under additional regularity assumptions (cf. [21, As. 1]). We include these assumptions as Assumption 4.1, which are valid even in our more general setting. In addition, we include assumptions that we will use to derive error estimates for the inexact Hessian applications.

Assumption 4.1. Let the following hold for problem $(\widehat{P})$:

(a) Let $Z_0 \subset Z_{\text{ad}}$ be open and bounded and let an open and bounded subset $U_0 \subset U$ exist, such that $\{u \in U \mid \exists\, z \in Z_0 \text{ s.t. } e(u, z) = 0\} \subseteq U_0$. Additionally, let an open and bounded subset $W_0^* \subset W^*$ exist, such that $\{p \in W^* \mid \exists\, z \in Z_0 \text{ s.t. } e_u(u(z), z)^* p = -J_u(u(z), z)\} \subseteq W_0^*$.

(b) There exist $0 < \underline{\sigma} \leq \overline{\sigma} < \infty$, such that the singular values[2] of $e_u(u, z)$ fulfill $\underline{\sigma} \leq \sigma_{\min}(e_u(u, z)) \leq \sigma_{\max}(e_u(u, z)) \leq \overline{\sigma}$ for all $u \in U_0$ and $z \in Z_0$.

(c) The mappings $u \mapsto e_u(u, z)$, $u \mapsto e_z(u, z)$, $u \mapsto J_u(u, z)$ and $u \mapsto J_z(u, z)$ are Lipschitz continuous on U_0 for all $z \in Z_0$ and their Lipschitz constants are independent of z.

As shown in [21, Sec. 5], Assumption 4.1 is fulfilled for discretized optimal control problems with parabolic PDE constraints. We will provide a more detailed discussion on this in Section 5, when applying the sketching method to an example problem with initial measure-valued control.

Next, we state some of the error estimates from [21, Prop. 4.1] without a proof. We use the notation $a \lesssim b$ to indicate that the inequality $a \leq cb$ holds up to a constant $c > 0$.

Lemma 4.2. *Let Z_0 be open and bounded, Assumption 4.1 hold, and $u(z) \in U_0$ be the state associated with z. For all $u \in U_0$, all $z \in Z_0$, and all $p \in W^*$ the following inequalities hold*

$$\underline{\sigma}\|u - u(z)\|_U \leq \|e(u, z)\|_W \leq \overline{\sigma}\|u - u(z)\|_U, \tag{4.2a}$$

$$\|p - p(z)\|_{W^*} \lesssim \|e(u, z)\|_W + \|e_u(u, z)^* p + J_u(u, z)\|_{U^*}, \tag{4.2b}$$

$$\|e_z(u, z)^* p + J_z(u, z) - \widehat{J}'(z)\|_{Z^*} \lesssim \|e(u, z)\|_W + \|e_u(u, z)^* p + J_u(u, z)\|_{U^*}. \tag{4.2c}$$

Remark 4.3.

(a) The constants in Lemma 4.2 solely depend on the constants from Assumption 4.1.

(b) From (4.2a) and (4.2c), it follows that

$$\|\widehat{J}'_r(z) - \widehat{J}'(z)\|_{Z^*} = \|e_z(\widetilde{u}, z)^* \widetilde{p} + J_z(\widetilde{u}, z) - \widehat{J}'(z)\|_{Z^*} \lesssim \|e(\widetilde{u}, z)\|_W \leq \overline{\sigma}\|\widetilde{u} - u(z)\|_U,$$

where we have used that $e_u(\widetilde{u}, z)^* \widetilde{p} + J_u(\widetilde{u}, z) = 0$.

Additionally, if $\widehat{J}$ is strongly convex, we have an error estimate for the control. The proof of this result is identical to [21, Th. 4.4] and is thus omitted.

[2] For the concept of singular values in Banach spaces, see e.g. [33].

Theorem 4.4. *Let Assumption 4.1 hold and $\widehat{J}$ be strongly convex on Z_0 with constant $\alpha > 0$. For given $z \in Z_0$, let $u(z) \in U_0$ be the associated state and let $\widetilde{u}$ be an approximation of $u(z)$. Furthermore, let the adjoint state $\widetilde{p} \in W_0^*$ associated with $\widetilde{u}$ solve*

$$e_u(\widetilde{u}, z)^* \widetilde{p} + J_u(\widetilde{u}, z) = 0.$$

If $\bar{z}$ is the solution to $(\widehat{P})$, then it holds that

$$\|z - \bar{z}\|_Z \lesssim \|e(\widetilde{u}, z)\|_W + \|\widehat{J}_r'(z)\|_{Z^*}.$$

The estimate in Theorem 4.4 is computable, since the quantities $\|e(\widetilde{u}, z)\|_W$ and $\|\widehat{J}_r'(z)\|_{Z^*}$ are computable. In the case that the inexactness is due to sketching with target rank r, we recover the result from [21, Th. 4.4].

Next, we derive an expression of the second-order derivatives and study the error induced by the approximation. We proceed as in [26, Sec. 1.6.5] using the Lagrangian approach. Define the Lagrangian $\mathcal{L} : U \times Z \times W^* \to \mathbb{R}$ as

$$\mathcal{L}(u, z, p) := J(u, z) + \langle p, e(u, z) \rangle_{W^*, W}. \tag{4.3}$$

Notice that for all $p \in W^*$, we have the identify

$$\widehat{J}(z) = J(u(z), z) + \langle p, \underbrace{e(u(z), z)}_{=0} \rangle_{W^*, W} = \mathcal{L}(u(z), z, p).$$

Then for $h_1 \in Z$, we obtain that

$$\langle \widehat{J}'(z), h_1 \rangle_{Z^*, Z} = \langle u'(z)^* \mathcal{L}_u(u(z), z, p) + \mathcal{L}_z(u(z), z, p), h_1 \rangle_{Z^*, Z}.$$

Differentiating again in direction $h_2 \in Z$ delivers

$$\begin{aligned} \langle \widehat{J}''(z) h_2, h_1 \rangle_{Z^*, Z} = {} & \langle \mathcal{L}_u(u(z), z, p), u''(z)(h_1, h_2) \rangle_{U^*, U} + \langle \mathcal{L}_{uu}(u(z), z, p) u'(z) h_2, u'(z) h_1 \rangle_{U^*, U} \\ & + \langle \mathcal{L}_{uz}(u(z), z, p) h_2, u'(z) h_1 \rangle_{U^*, U} + \langle \mathcal{L}_{zu}(u(z), z, p) u'(z) h_2, h_1 \rangle_{Z^*, Z} \\ & + \langle \mathcal{L}_{zz}(u(z), z, p) h_2, h_1 \rangle_{Z^*, Z}. \end{aligned}$$

It is easy to verify that for the adjoint state $p = p(z)$ it holds $\mathcal{L}_u(u(z), z, p(z)) = 0$. Consequently, the first summand vanishes. Furthermore, we have $u'(z) = -e_u(u(z), z)^{-1} e_z(u(z), z)$, which can be derived by implicitly differentiating $e(u(z), z) = 0$ with respect to z. In view of minimizing the necessary storage, we refrain from storing the full Hessian, and examine how to apply the Hessian to an arbitrary vector $v \in Z$. Altogether, we arrive at

$$\begin{aligned} \widehat{J}''(z) v = {} & e_z(u(z), z)^* e_u(u(z), z)^{-*} \mathcal{L}_{uu}(u(z), z, p(z)) e_u(u(z), z)^{-1} e_z(u(z), z) v \\ & - e_z(u(z), z)^* e_u(u(z), z)^{-*} \mathcal{L}_{uz}(u(z), z, p(z)) v \\ & - \mathcal{L}_{zu}(u(z), z, p(z)) e_u(u(z), z)^{-1} e_z(u(z), z) v + \mathcal{L}_{zz}(u(z), z, p(z)) v. \end{aligned}$$

In abbreviated notation, this is equivalent to

$$\widehat{J}''(z) v = -e_z^* \underbrace{\left(e_u^{-*} \left(\mathcal{L}_{uu} \left(-e_u^{-1} e_z v \right) + \mathcal{L}_{uz} \right) \right)}_{=: q(z; v)} + \mathcal{L}_{zu} \underbrace{\left(-e_u^{-1} e_z v \right)}_{=: w(z; v)} + \mathcal{L}_{zz} v, \tag{4.4}$$

which motivates [34, Alg. 2].

Remark 4.5. In the context of the dynamic optimization problem (1.1), we compress the adjoint variable $p \in W^*$ and the state sensitivity variable $w \in U$ to evaluate $\widehat{J}''(z)v$. However, for simplicity of notation, we only derive the error estimates for the approximate state $\widetilde{u}$. Further compression errors related to the other variables can be introduced and the proofs can be directly extended.

Next, we estimate the approximation error in the application of Hessian under the following additional assumptions.

Assumption 4.6. Let the second derivatives of the Lagrangian (4.3), i.e., $\mathcal{L}_{uu}(u, z, p)$, $\mathcal{L}_{uz}(u, z, p)$, and $\mathcal{L}_{zz}(u, z, p)$, be Lipschitz continuous on $U_0 \times Z_0 \times W_0^*$ with respect to the first and third arguments, and suppose that the Lipschitz constants are independent of the second argument. Here, U_0, Z_0 and W_0^* are defined in Assumption 4.1.

We will see in Section 5 that Assumption 4.6 holds for the discretized optimal control problems with quadratic objective function and linear PDE constraint. Next, we discuss the approximation errors that occur in the application of the Hessian due to the inexact state. This will enable us to analyze the inexact second-order methods introduced in the subsequent sections.

Theorem 4.7. *Let Assumptions 4.1 and 4.6 hold, and let $u(z) \in U_0$ be the state associated with control $z \in Z_0$. For all $z \in Z_0$, $v \in Z$, $u \in U_0$, $w \in U$, and $p, q \in W_0^*$ the following inequalities hold:*

$$\|w - w(z; v)\|_U \lesssim (1 + \|v\|_Z) \|e(u, z)\|_W + \|e_u(u, z) w + e_z(u, z) v\|_W, \tag{4.5a}$$

$$\|q - q(z; v)\|_{W^*} \lesssim (1 + \|v\|_Z) \|e(u, z)\|_W + (1 + \|v\|_Z) \|e_u(u, z)^* p + J_u(u, z)\|_{U^*}$$

$$
\begin{aligned}
&+ \|e_u(u,z)w + e_z(u,z)v\|_W \\
&+ \|e_u(u,z)^* q - \mathcal{L}_{uu}(u,z,p)w - \mathcal{L}_{uz}(u,z,p)v\|_{U^*}, \qquad (4.5b)\\
\| - e_z(u,z)^* q + \mathcal{L}_{zu}(u,z,p)w + \mathcal{L}_{zz}(u,z,p)v - \hat{J}''(z)v\|_{Z^*} &\\
\lesssim (1 + \|v\|_Z)\|e(u,z)\|_W + (1 + \|v\|_Z)\|e_u(u,z)^* p + J_u(u,z)\|_{U^*} &\\
+ \|e_u(u,z)w + e_z(u,z)v\|_W &\\
+ \|e_u(u,z)^* q - \mathcal{L}_{uu}(u,z,p)w - \mathcal{L}_{uz}(u,z,p)v\|_{U^*}. \qquad (4.5c)
\end{aligned}
$$

Remark 4.8. All constants in Theorem 4.7 solely depend on the constants in Assumptions 4.1 and 4.6.

The proof of Theorem 4.7 is straight forward, see Appendix A. For related arguments, with different notations, we refer to [35, Sec. 3]. These results can be directly applied to establish estimates for the approximation of Hessian application from (4.4) as we see next.

Corollary 4.9. *Under the assumptions of Theorem 4.7, the following estimates hold*

$$
\begin{aligned}
\|\hat{J}_r''(z)v - \hat{J}''(z)v\|_{Z^*} &= \| - e_z(\widetilde{u},z)^*\widetilde{q} + \mathcal{L}_{zu}(\widetilde{u},z,\widetilde{p})\widetilde{w} + \mathcal{L}_{zz}(\widetilde{u},z,\widetilde{p})v - \hat{J}''(z)v\|_{Z^*}\\
&\lesssim (1 + \|v\|)\|e(\widetilde{u},z)\|_W\\
&\lesssim (1 + \|v\|)\|\widetilde{u} - u(z)\|_U.
\end{aligned}
$$

Proof. This is a consequence of (4.5c) and the facts that $0 = e_u(\widetilde{u},z)^*\widetilde{p} + J_u(\widetilde{u},z) = e_u(\widetilde{u},z)\widetilde{w} + e_z(\widetilde{u},z)v = e_u(\widetilde{u},z)^*\widetilde{q} - \mathcal{L}_{uu}(\widetilde{u},z,\widetilde{p})\widetilde{w} - \mathcal{L}_{uz}(\widetilde{u},z,\widetilde{p})v$. Additionally, from (4.2a) we have that $\|e(\widetilde{u},z)\|_W \le \overline{\sigma}\|\widetilde{u} - u(z)\|_U$. □

Remark 4.10. Corollary 4.9 implies that the error in the Hessian application is controlled by the state compression error. When the adjoint variable p and the state sensitivity variable w are also compressed, the respective error terms in (4.5c) will be nonzero, in contrast to Corollary 4.9. However, these errors can be easily accounted for in the final estimate.

4.2. Adaptive inexact semismooth Newton's method

We are now in position to verify the critical conditions in Assumption 3.2(b) and (c). By adaptively reducing the approximation error (e.g., due to sketching), the resulting Algorithm 2 is shown to be provably convergent. In contrast to Algorithm 1, we include the computation of a step length t^k in Algorithm 2 as discussed in Remark 3.5.

Algorithm 2 Adaptive Inexact Semismooth Newton's Method

Require: The initial guess $x^0 = (z^0, \lambda^0, \mu^0) \in Z \times H^* \times G$, and $\beta > 0$.

for $k = 0, 1, \ldots$ **do**
  Solve the state equation for $u(z^k)$ and replace it with an approximation (e.g., via sketching) to obtain $\widetilde{u}^k$, simultaneously compute the error $\|e(\widetilde{u}^k, z^k)\|_Y$.
  Solve the adjoint equation with input $\widetilde{u}^k$ for $\widetilde{p}^k$, simultaneously compute $\widetilde{F}_k(x^k)$.
  if $\|e(\widetilde{u}^k, z^k)\|_W \le \|\widetilde{F}_k(x^k)\|_Y^{1+\beta}$ is not fulfilled **then**
    Adjust the approximation to decrease the residual $\|e(\widetilde{u}^k, z^k)\|_W$, and go to 2.
  end if
  Choose $\widetilde{M}^k$ as an approximation to $M^k \in \partial^* F(x^k)$ boundedly invertible in $L(X, Y)$.
  Obtain $s^k \in X$ by solving $\widetilde{M}^k s^k = -\widetilde{F}_k(x^k)$.
  Determine stepsize t^k.
  Set $x^{k+1} = x^k + t^k s^k$.
end for

The adaptive inexact semismooth Newton's method, Algorithm 2, ensures that

$$
\|e(\widetilde{u}^k, z^k)\|_W \le \|\widetilde{F}_k(x^k)\|_Y^{1+\beta}, \qquad \beta > 0, \qquad (4.6)
$$

holds at each iteration k by adjusting the state approximation $\widetilde{u}^k$ to decrease the compression error. We make the following assumption to guarantee that it is possible to sufficiently decrease the approximation error.

Assumption 4.11. The accuracy of $\widetilde{u}^k$ of $u(z^k)$ can be increased until the error, $\|u(z^k) - \widetilde{u}^k\|_U$, vanishes.

A motivation for (4.6) is provided in the result below. In the context of dynamic optimization, the computation of the residual $\|e(\widetilde{u}^k, z^k)\|_W$ in Algorithm 2 is performed sequentially, as described in Algorithm 4. Hence, the adaptive condition (4.6) can be verified within the realm of efficient storage.

Theorem 4.12. *Consider problem (P). Let F be given as in (3.5) and let $\bar{x} = (\bar{z}, \bar{\lambda}, \bar{\mu}) \in Z_{\mathrm{ad}} \times H^* \times G$ solve (3.1). Let Assumptions 3.2(a), 4.1, 4.6, and 4.11 hold, and additionally let there be a constant C_M, such that*

$$\|\widetilde{M}^k\|_{\mathscr{L}(X,Y)} \le C_M \qquad \forall k. \tag{4.7}$$

Then, Algorithm 2 either:

(i) Terminates with $x^* = x^k = \bar{x}$, or
(ii) Generates a sequence $\{x^k\}$ that converges q-superlinearly to $\bar{x}$ in X.

Proof. This result follows from Theorem 3.4 if we can show that Assumption 3.2(b) and (c) are fulfilled, i.e. $\|\Delta^k s^k\|_Y = o(\|s^k\|_X)$ and $\|\epsilon^k\|_Y = o(\|\widetilde{F}(x^k)\|_Y)$ as $\|s^k\|_X \to 0$ and $\|\widetilde{F}(x^k)\|_Y \to 0$, where $\Delta^k = \widetilde{M}^k - M^k$ and $\epsilon^k = \widetilde{F}_k(x^k) - F(x^k)$. By Remark 4.3 and Corollary 4.9 with $v = s^k$, we have that

$$\|\epsilon^k\|_Y \lesssim \|e(\widetilde{u}^k, z^k)\|_W \quad \text{and} \quad \|\Delta^k s^k\|_Y \lesssim (1 + \|s^k\|_X)\|e(\widetilde{u}^k, z^k)\|_W.$$

Furthermore, from the Newton step and (4.7) we have

$$\|\widetilde{F}_k(x^k)\|_Y = \|\widetilde{M}^k s^k\|_Y \le C_M \|s^k\|_X,$$

and we can directly deduce $\|\widetilde{F}_k(x^k)\|_Y^{1+\beta} \lesssim \|s^k\|_X^{1+\beta}$. Once (4.6) holds true, we have

$$\|\epsilon^k\|_Y \lesssim \|\widetilde{F}_k(x^k)\|_Y^{1+\beta}, \qquad \text{and} \qquad \|\Delta^k s^k\|_Y \lesssim (1 + \|s^k\|_X)\|s^k\|_X^{1+\beta},$$

with constants independent of k. Finally, with $\beta > 0$, as $\|\widetilde{F}_k(x^k)\|_Y \to 0$ and $\|s^k\|_X \to 0$, it will hold $\frac{\|\epsilon^k\|_Y}{\|\widetilde{F}_k(x^k)\|_Y} \to 0$ and $\frac{\|\Delta^k s^k\|_Y}{\|s^k\|_X} \to 0$, i.e. Assumption 3.2(b) and (c) are fulfilled. This completes the proof. □

Remark 4.13 (*Krylov Methods and Stepsizes*).

(a) To ameliorate the cost of storing the full matrix $\widetilde{M}^k$, we can employ Krylov methods (e.g., GMRES [36]) to solve the semismooth Newton step (Step 8 in Algorithm 2), which only require the action of $\widetilde{M}^k$ onto a vector. Such methods introduce an additional source of inexactness in Algorithm 2, which for every k, we can write as

$$\widetilde{M}^k s^k = -(\widetilde{F}_k(x^k) + \epsilon^k_{\mathrm{Krylov}}).$$

Under the additional assumption that this error, $\epsilon^k_{\mathrm{Krylov}}$ is $o(\|\widetilde{F}_k(x^k)\|_Y)$ as $\|\widetilde{F}_k(x^k)\|_Y \to 0$, we maintain the convergence result, Theorem 3.4. This adaptivity can be ensured by setting the desired tolerance of the Krylov space method to be less or equal $\|\widetilde{F}_k(x^k)\|_Y^{1+\beta}$.

(b) Numerically, we notice that for small stepsize t^k, adjusting the compression parameters to decrease the compression error can accelerate convergence. However, such an adjustment depends on the available storage.

(c) Notice that for dynamic optimization problems, and if the inexactness arise from randomized sketching, one generates the sketch of the state in Step 2 of Algorithm 2 using Algorithm 3 from Appendix. The reconstruction of state (needed for adjoint solves in Step 3) and calculation of sketching error (Step 4–5) is done using Algorithm 4.

5. Example problem: Initial measure control

Let $\Omega \subset \mathbb{R}^d, 1 \le d \le 3$ be an open, connected and bounded set with Lipschitz boundary Γ, and let $T > 0$ denote the final time. We denote the space–time domain by $Q := \Omega \times (0, T)$ and assume that $u_{\mathrm{d}} \in L^2(Q)$ is given. In addition, denote by $\mathcal{M}(\overline{\Omega})$ the space of real and regular Borel measures, and recall that $\mathcal{M}(\overline{\Omega})$ is the dual space of $C(\overline{\Omega})$, the space of continuous functions, cf. [37, p. 130]. We endow $\mathcal{M}(\overline{\Omega})$ with the total variation norm

$$\|z\|_{\mathcal{M}(\overline{\Omega})} = \sup_{\|\phi\|_{C(\overline{\Omega})} \le 1} \int_{\overline{\Omega}} \phi \, \mathrm{d}z.$$

To illustrate the benefits of sketching, we consider an optimal control problem with bounded initial measure control. This problem is inspired by [38, Rem. 2.11] and in its reduced form, it is given by

$$\begin{aligned} &\min_z \left\{ \hat{J}(z) := \frac{1}{2} \int_Q \chi_{(t_0,T)\times\Omega} |u(z) - u_{\mathrm{d}}|^2 \right\} \\ &\text{subject to} \quad z \in \mathcal{Z}_\alpha := \{ z \in \mathcal{M}(\overline{\Omega}) \mid \|z\|_{\mathcal{M}(\overline{\Omega})} \le \alpha \} \end{aligned} \tag{P_α}$$

with $\alpha > 0$ and $t_0 > 0$, where the state $u = u(z)$, in a very-weak sense [38], fulfills

$$\partial_t u - \Delta u = 0 \qquad \text{in } Q := \Omega \times (0, T), \tag{5.1a}$$

$$\partial_\eta u = 0 \qquad \text{on } \Sigma := \Gamma \times (0, T), \tag{5.1b}$$

$$u(0) = z \qquad \text{in } \Omega. \tag{5.1c}$$

The state Eq. (5.1) admits a unique solution $u \in L^r(0,T;W_0^{1,p}(\Omega)) \cap C((0,T];L^2(\Omega))$ for all $p,r \in [1,2)$ that fulfill $(2/r)+(d/p) > d+1$, see [38, Th. 2.2]. Also, following [38, Th. 2.4], we see that (P_α) has a unique solution.

The full state, at all time steps, is needed to evaluate the adjoint variable p at $t=0$, which enters in the optimality conditions. Owing to the structure of the problem, sketching will be highly beneficial. We follow [38, Th. 2.5 &Rem. 2.11] to derive the optimality conditions first.

Lemma 5.1. *Let $u_d \in L^q(0,T;L^2(\Omega))$ with $q \in [1,\infty)$ and let $\bar{z}$ solve (P_α) with associated state $\bar{u}$. Then the adjoint state $\bar{p} \in L^2(0,T;H^1(\Omega)) \cap C([0,T)\times\overline{\Omega})$, in a weak sense [38] solves*

$$-\partial_t \bar{p} - \Delta \bar{p} = (\bar{u}-u_d)\chi_{(t_0,T)\times\Omega} \qquad \text{in } Q, \tag{5.2a}$$

$$\partial_\eta \bar{p} = 0 \qquad \text{on } \Sigma, \tag{5.2b}$$

$$\bar{p}(T) = 0 \qquad \text{in } \Omega. \tag{5.2c}$$

If $\int_Q \chi_{(t_0,T)\times\Omega}(\bar{u}-u_d)\,dx\,dt \neq 0$, then the following conditions hold:

$$\|\bar{z}\|_{\mathcal{M}(\overline{\Omega})} = \alpha,$$
$$\operatorname{supp}(\bar{z}^+) \subset \{x \in \overline{\Omega} \mid \bar{p}(x,0) = -\|\bar{p}(0)\|_{C(\overline{\Omega})}\},$$
$$\operatorname{supp}(\bar{z}^-) \subset \{x \in \overline{\Omega} \mid \bar{p}(x,0) = +\|\bar{p}(0)\|_{C(\overline{\Omega})}\},$$

where $\bar{z} = \bar{z}^+ - \bar{z}^-$ is the Jordan decomposition of $\bar{z}$. Conversely, if $\bar{z} \in \mathcal{Z}_\alpha$ fulfills the above properties, then $\bar{z}$ solves (P_α).

Proof. Since $t_0 > 0$, the proof follows along the arguments in [38, Theorem 2.5].

Remark 5.2. The assumption $\int_Q \chi_{(t_0,T)\times\Omega}(\bar{u}-u_d)\,dx\,dt \neq 0$ is meaningful, since it simply means that the desired state $u_d \in L^2(Q)$ is not in the reachable set $\mathcal{R} := \{u(z) \mid \|z\|_{\mathcal{M}} \leq \alpha\}$.

To discretize (5.1), we choose the discrete state and test space to be piecewise linear and continuous in space, and piecewise constant in time, which leads to an implicit Euler time stepping scheme for the parabolic PDE. We employ a variational discretization space for the control, i.e., the control is not discretized. Variational discretization was introduced in [39] and has been used in [40] for a related problem with initial measure control, and in [41], which considers a measure norm regularization term, instead of the bound on the measure control. Following [40, Th. 6], the resulting problem can be shown to have at least one solution, and the optimality conditions can be derived in the same way as in the continuous setting (cf. [40, Th. 11]). In particular, it holds

$$\operatorname{supp}(\bar{z}^+) \subset \{x \in \overline{\Omega} \mid \bar{\mathbf{p}}(x,0) = -\|\bar{\mathbf{p}}(0)\|_{C(\overline{\Omega})}\} \tag{5.3a}$$

$$\operatorname{supp}(\bar{z}^-) \subset \{x \in \overline{\Omega} \mid \bar{\mathbf{p}}(x,0) = +\|\bar{\mathbf{p}}(0)\|_{C(\overline{\Omega})}\}, \tag{5.3b}$$

where $\bar{\mathbf{p}}$ is the discrete adjoint state associated with the solution $\bar{z}$. Since $\bar{\mathbf{p}}(0)$ is a piecewise linear function, it will attain minima and maxima at grid points in the generic case, i.e., we assume that $\bar{\mathbf{p}}(0)$ is not constant on spatial elements. We immediately deduce

$$\operatorname{supp}(\bar{z}) \subset \{x_j\}_{j=1}^{n_s},$$

with x_j being the grid points. Consequently, the optimal control has the following implicit discrete structure

$$\bar{\mathbf{z}} \in \mathcal{Z}_h := \operatorname{span}\{\delta_{x_j} \mid 1 \leq j \leq n_s\}.$$

This is typical of the variational discretization—a natural discretization for the control space is induced by the chosen test space discretization. Furthermore, if we restrict our control space to $\mathcal{Z}_\alpha \cap \mathcal{Z}_h$ the resulting finite-dimensional problem admits a unique solution.

In order to implement the problem, we need discrete versions of the state and adjoint equations. Let $\mathbf{M}$ and $\mathbf{A}$ be the spatial mass and stiffness matrices, respectively, and τ the equidistant time step size in $(0,T)$. We define the following space–time matrices

$$\mathscr{M} := \begin{pmatrix} \tau\mathbf{M} & & \\ & \ddots & \\ & & \tau\mathbf{M} \end{pmatrix}, \qquad \mathscr{S} := \begin{pmatrix} \mathbf{M}+\tau\mathbf{A} & & & \\ -\mathbf{M} & \ddots & & \\ & \ddots & \ddots & \\ & & -\mathbf{M} & \mathbf{M}+\tau\mathbf{A} \end{pmatrix}.$$

The state equation, which determines $\mathbf{u}(\mathbf{z})$, can be written as

$$\mathscr{S}\mathbf{u} = \begin{pmatrix} \mathbf{z} & \cdots & 0 \end{pmatrix}^\top, \tag{5.4}$$

where the right-hand side does not contain the spatial mass matrix $\mathbf{M}$, since evaluating piecewise linear functions against dirac measures delivers an identity matrix. The associated adjoint equation reads

$$\mathscr{S}^\top \mathbf{p} = \mathscr{M}(\mathbf{u}_d - \mathbf{u}).$$

Notice that in the discrete setting the state is a vector $\mathbf{u} \in \mathbb{R}^{n_s n_t}$. However, to apply the sketching technique, we reshape the vector $\mathbf{u}$ into the state matrix $\mathbf{U} \in \mathbb{R}^{n_s \times n_t}$, so that every column represents a time instance, cf. Section 2. Recall that we never form $\mathbf{U}$.

We are now ready to state the discrete optimization problem

$$\begin{aligned} &\min_{\mathbf{z}\in\mathcal{Z}_h} \left\{ \hat{J}_\sigma(\mathbf{z}) := \frac{1}{2}(\mathbf{u}(\mathbf{z}) - \mathbf{u}_\mathrm{d})^\top \mathscr{M}(\mathbf{u}(\mathbf{z}) - \mathbf{u}_\mathrm{d}) \right\} \\ &\text{subject to} \qquad \sum_{i=1}^{n_s} |\mathbf{z}_i| - \alpha \le 0. \end{aligned} \tag{$P_{\alpha,\sigma}$}$$

Here, $\sigma = (h, \tau)$ represents the space–time discretization. Note that Assumption 4.1 holds for the discretized problem ($P_{\alpha,\sigma}$) since the objective function is quadratic and the discretized state equation is linear. To tackle the non-differentiable absolute value, we consider a decomposition of the control, namely $\mathbf{z} = \mathbf{z}^+ - \mathbf{z}^-$ with $\mathbf{z}^+, \mathbf{z}^- \ge 0$. Furthermore, we enforce $\mathbf{z}_i^+ \mathbf{z}_i^- = 0$ for all $i = 1, \dots, n_s$ by adding a penalty term to the target functional. Let us remark that the introduction of this constraint is essential to preserve uniqueness of solutions. For example, if $\sum_{i=1}^{n_s} |\mathbf{z}_i| - \alpha < 0$, there are infinitely many, different decompositions $\mathbf{z}_i = \mathbf{z}_i^+ - \mathbf{z}_i^-$, such that $\sum_{i=1}^{n_s} (\mathbf{z}_i^+ + \mathbf{z}_i^-) - \alpha \le 0$ is still fulfilled. We can then write the problem as follows

$$\begin{aligned} &\min_{\mathbf{z}^+,\mathbf{z}^-\in\mathcal{Z}_h} \left\{ \hat{J}_{\sigma,\gamma}(\mathbf{z}^+, \mathbf{z}^-) = \frac{1}{2}(\mathbf{u}(\mathbf{z}^+ - \mathbf{z}^-) - \mathbf{u}_\mathrm{d})^\top \mathscr{M}(\mathbf{u}(\mathbf{z}^+ - \mathbf{z}^-) - \mathbf{u}_\mathrm{d}) + \gamma(\mathbf{z}^+)^\top \mathbf{z}^- \right\} \\ &\text{subject to} \qquad \sum_{i=1}^{n_s} (\mathbf{z}_i^+ + \mathbf{z}_i^-) - \alpha \le 0, \quad -\mathbf{z}_i^+ \le 0 \quad \forall\, i, \quad \text{and} \quad -\mathbf{z}_i^- \le 0 \quad \forall\, i. \end{aligned} \tag{$P_{\alpha,\sigma,\gamma}$}$$

Here $\gamma > 0$ is a penalty parameter, which in practice is updated using path following (cf. [25, Ch. 8]). It can be proven that ($P_{\alpha,\sigma,\gamma}$) is an equivalent reformulation of ($P_{\alpha,\sigma}$), if the penalty parameter γ is large enough, see [42, Th. 4.17].

Next, we set up the Lagrangian functional with multipliers $\mu_1 \in \mathbb{R}$ and $\mu_2, \mu_3 \in \mathbb{R}^{n_s}$

$$\begin{aligned} \mathscr{L}(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) := {} & \frac{1}{2}(\mathbf{u}(\mathbf{z}^+ - \mathbf{z}^-) - \mathbf{u}_\mathrm{d})^\top \mathscr{M}(\mathbf{u}(\mathbf{z}^+ - \mathbf{z}^-) - \mathbf{u}_\mathrm{d}) + \gamma(\mathbf{z}^+)^\top \mathbf{z}^- \\ & + \mu_1 \Big(\sum_{i=1}^{n_s} (\mathbf{z}_i^+ + \mathbf{z}_i^-) - \alpha \Big) - \sum_{i=1}^{n_s} \mu_{2,i} \mathbf{z}_i^+ - \sum_{i=1}^{n_s} \mu_{3,i} \mathbf{z}_i^-. \end{aligned}$$

Now, following [40, Sect. 4], we form the KKT system and reformulate the complementarity conditions equivalently into equations with some $\kappa > 0$. This yields the optimality system

$$F(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) = \begin{pmatrix} \partial_{\mathbf{z}^+} \mathscr{L}(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) \\ \partial_{\mathbf{z}^-} \mathscr{L}(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) \\ \max\left\{0, \mu_1 + \kappa \left(\sum_{i=1}^{n_s} (\mathbf{z}_i^+ + \mathbf{z}_i^-) - \alpha\right)\right\} - \mu_1 \\ \max\{0, \mu_2 - \kappa \mathbf{z}^+\} - \mu_2 \\ \max\{0, \mu_3 - \kappa \mathbf{z}^-\} - \mu_3 \end{pmatrix} = 0, \tag{5.5}$$

where

$$\begin{aligned} \partial_{\mathbf{z}^+} \mathscr{L}(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) &= -\mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} + \gamma \mathbf{z}^- + \begin{pmatrix} \mu_1 & \dots & \mu_1 \end{pmatrix}^\top - \mu_2, \\ \partial_{\mathbf{z}^-} \mathscr{L}(\mathbf{z}^+, \mathbf{z}^-, \mu_1, \mu_2, \mu_3) &= \mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} + \gamma \mathbf{z}^+ + \begin{pmatrix} \mu_1 & \dots & \mu_1 \end{pmatrix}^\top - \mu_3. \end{aligned}$$

Let us define the following sets

$$\begin{aligned} \mathcal{A}_1 &:= \left\{ \mu_1 + \kappa \left(\sum_{i=1}^{n_s} (\mathbf{z}_i^+ + \mathbf{z}_i^-) - \alpha \right) \ge 0 \right\}, & \mathcal{I}_1 &:= \mathbb{R} \setminus \mathcal{A}_1, \\ \mathcal{A}_2 &:= \left\{ \mu_2 - \kappa \mathbf{z}^+ \ge 0 \right\}, & \mathcal{I}_2 &:= \Omega \setminus \mathcal{A}_2, \\ \mathcal{A}_3 &:= \left\{ \mu_3 - \kappa \mathbf{z}^- \ge 0 \right\}, & \mathcal{I}_3 &:= \Omega \setminus \mathcal{A}_3. \end{aligned}$$

For the generalized Jacobian of F, we select $\partial_x(\max\{0, g(x)\}) = \partial_x g(x)$ if $g(x) = 0$, so that

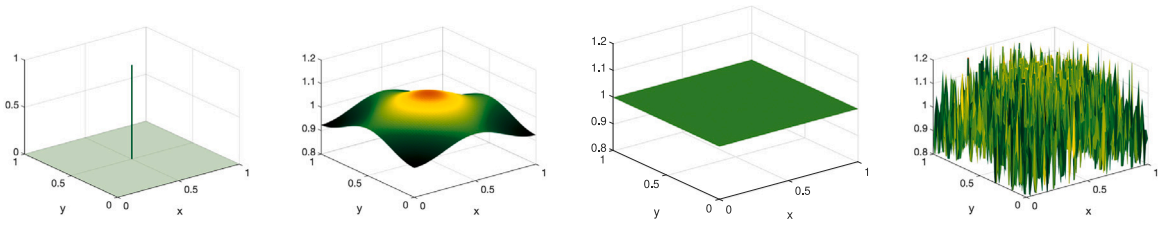
$$DF = \begin{pmatrix} -\partial_{\mathbf{z}^+} \mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} & -\partial_{\mathbf{z}^-} \mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} + \gamma \mathbf{I} & \mathbf{1}_{n_s} & -\mathbf{I} & 0 \\ \partial_{\mathbf{z}^+} \mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} + \gamma \mathbf{I} & \partial_{\mathbf{z}^-} \mathbf{p}(\mathbf{z}^+ - \mathbf{z}^-)_{|t=0} & \mathbf{1}_{n_s} & 0 & -\mathbf{I} \\ \begin{pmatrix} \kappa & \dots & \kappa \end{pmatrix}_{\mathcal{A}_1} & \begin{pmatrix} \kappa & \dots & \kappa \end{pmatrix}_{\mathcal{A}_1} & -1_{\mathcal{I}_1} & 0 & 0 \\ -\kappa \mathbf{I}_{\mathcal{A}_2} & 0 & 0 & -\mathbf{I}_{\mathcal{I}_2} & 0 \\ 0 & -\kappa \mathbf{I}_{\mathcal{A}_3} & 0 & 0 & -\mathbf{I}_{\mathcal{I}_3} \end{pmatrix}, $$

where all identity matrices $\mathbf{I}$ are of size $n_s \times n_s$ and matrices indexed with sets are only non-zero on those sets, e.g., $1_{\mathcal{I}_1}$=1 if $\mathcal{A}_1 = \emptyset$.

We solve the generalized Newton system using GMRES without preconditioning. We use the backtracking linesearch: $t^k = 0.5^j$ where j is the smallest nonnegative integer for which

$$\|F(z^k + t^k s^k)\|_2 \le (1 - 10^{-4} t^k) \|F(z^k)\|_2 \tag{5.6}$$

holds. To avoid numerical issues, we enforce that $j \le 10$. As we will see below, the performance of sketched methods is almost the same as without sketching. We set the maximum number of iterations to 300. The relative stopping tolerance for GMRES is 10^{-6}. We terminate our semismooth Newton method if $\|F(x^k)\|_Y \le 10^{-8}$.



(a) z_{true} (b) $u(z_{\text{true}})_{|t=0.1}$ (c) $u(z_{\text{true}})_{|t=T}$ (d) $u_{d|t=0.1}$

Fig. 1. Generation of the desired state u_{d}. From left to right: True solution $z_{\text{true}} = \delta_{0.5,0.5}$, the associated state $u(z_{\text{true}})$ computed using Fourier modes evaluated at $t = 0.1$ and $t = T = 2$. Last panel shows u_{d} at $t = 0.1$, which is generated by adding normally distributed random noise to $u(z_{\text{true}})$.

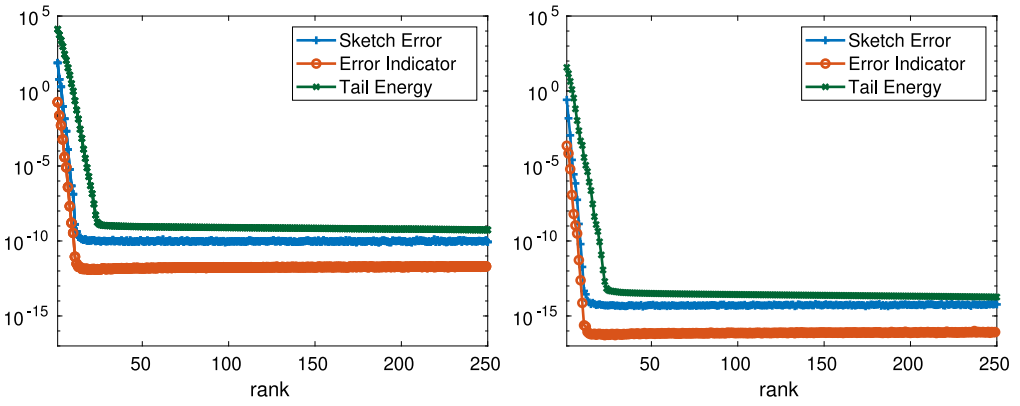


Fig. 2. The two panels display the 'sketch error' $\|\mathbf{u} - \widetilde{\mathbf{u}}\|_2$, the error indicator $\|\mathbf{c}(\widetilde{\mathbf{u}}, \mathbf{z})\|_2$, and the tail energy $\tau_{r+1}(\mathbf{u})$ for random input $(\mathbf{z}, \mu_1, \mu_2, \mu_3)$ (left) and the optimal solution $(\bar{\mathbf{z}}, \bar{\mu}_1, \bar{\mu}_2, \bar{\mu}_3)$ (right) with respect to rank r. These results corroborate the error bound (2.6). To account for randomness, the average over 10 realizations is displayed. Recall that $k = 2r + 1$ is the rank of the sketch $\widetilde{\mathbf{u}}$, therefore the sketch error curve is below the tail energy.

5.1. Nonnegative measures

An interesting special case arises, if only nonnegative measures are considered in problem (P_α), i.e., $z \in \mathcal{Z}_\alpha^+ := \{z \in \mathcal{M}^+(\overline{\Omega}) \,|\, \|z\|_{\mathcal{M}(\overline{\Omega})} \leq \alpha\}$. Here, $\mathcal{M}^+(\overline{\Omega})$ is the subspace of $\mathcal{M}(\overline{\Omega})$ that contains all nonnegative measures. This problem has been analyzed in [38, Sect. 3] and its variational discretization can be found in [40]. We therefore briefly discuss how the variational discrete problem $(P_{\alpha,\sigma,\gamma})$ simplifies when nonnegative measure controls are considered. For more details, we refer to the aforementioned references. First of all, we do not need to decompose the control $\mathbf{z}$ into positive and negative parts since $\mathbf{z} = \mathbf{z}^+$. Hence, no penalization is needed, i.e., $\gamma = 0$. Furthermore, it holds that $|\mathbf{z}_i| = \mathbf{z}_i^+$ for all $i = 1, \dots, n_s$, and we only have one nonnegativity constraint, i.e. $-\mathbf{z}_i^+ \leq 0$ for all i. Consequently, the optimality system reduces to the first, third and fourth equations in F, cf. (5.5). That is, $\mathcal{A}_3$ is not needed.

5.2. Numerical results

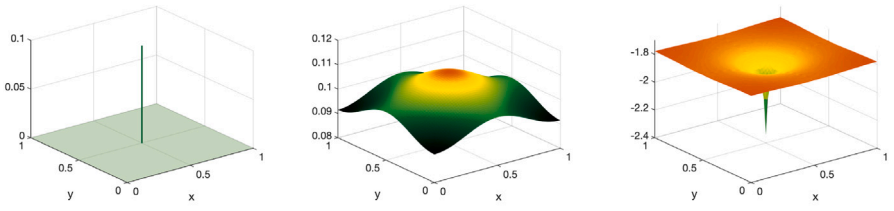
Consider the domain $\Omega = (0, 1)^2$ and the final time $T = 2$.

Nonnegative measures

We let $n_s = 64^2$ (spatial nodes) and $n_t = 501$ (time steps), so that $n_s n_t = 2,052,096$. Furthermore, we fix $\alpha = 0.1$. We generate a desired state u_{d} by solving the state equation with the initial measure control $z_{\text{true}} = \delta_{(0.5,0.5)}$ and adding normally distributed random noise (mean 0 and standard deviation 0.1). This data is depicted in Fig. 1.

Fig. 2 shows the sketch error $\|\mathbf{u} - \widetilde{\mathbf{u}}\|_2$, the error indicator $\|\mathbf{c}(\widetilde{\mathbf{u}}, \mathbf{z})\|_2$ and the tail energy of the full state $\mathbf{u}$ as functions of the sketching target rank r for random inputs $(\mathbf{z}, \mu_1, \mu_2)$ (left) and the optimal solution $(\bar{\mathbf{z}}, \bar{\mu}_1, \bar{\mu}_2)$ (right). These results corroborate the error bound (2.6). We note that the residual-based error indicator is smaller than the sketching error because we did not scale it by the constant $1/\underline{\sigma}$ as in Lemma 4.2. The left panel shows an exponential decay, which is an indication that sketching will work well, as shown on the right panel. Indeed, it is possible to obtain machine precision accuracy with relatively small r.

Fig. 3 shows the optimal solution. The support of $\bar{z}$ coincides with the support of z_{true} (see Fig. 1), and it holds that $\|\bar{z}\|_{\mathcal{M}(\overline{\Omega})} = \alpha = 0.1$. Furthermore, the optimal solution clearly fulfills the support subset condition (5.3a). Similar plots are obtained whether we consider the full problem without sketching or with sketching using adaptive rank or fixed rank with $r > 2$. Consequently, we only display one of these cases.



(a) $\bar{z}$ (b) $u(\bar{z})_{|t=0.1}$ (c) $p(\bar{z})_{|t=0}$

Fig. 3. Solution for $\alpha = 0.1$. From left to right: The optimal control $\bar{\mathbf{z}}$, the optimal state $\bar{\mathbf{u}} = \mathbf{u}(\bar{\mathbf{z}})$ evaluated at $t = 0.1$, and the optimal adjoint state $\bar{\mathbf{p}} = \mathbf{p}(\bar{\mathbf{z}})$ evaluated at $t = 0$. The optimality condition, (5.3a) is visibly satisfied.

Table 1
The performance of semismooth Newton's method for nonnegative initial measure control. Three cases are considered: Without sketching, sketching with fixed rank, and sketching with adaptive rank. The columns of the table correspond to the rank, the number of iterations, the final objective function value, the final residual $\|F\|_2$ and the compression factor. The sketched methods perform comparably to the unsketched method, but require significantly less memory.

Rank	Iterations	Objective	Residual	Compression
1*	300	791.4969	4.0e−01	148
2*	300	13.5375	6.4e−02	89
3	56	23.7671	1.4e−09	63
4	48	23.7671	4.0e−09	49
5	42	23.7671	5.5e−09	40
10	47	23.7671	3.0e−10	21
Adaptive	49	23.7671	5.1e−09	49
Full	47	23.7671	3.0e−10	–

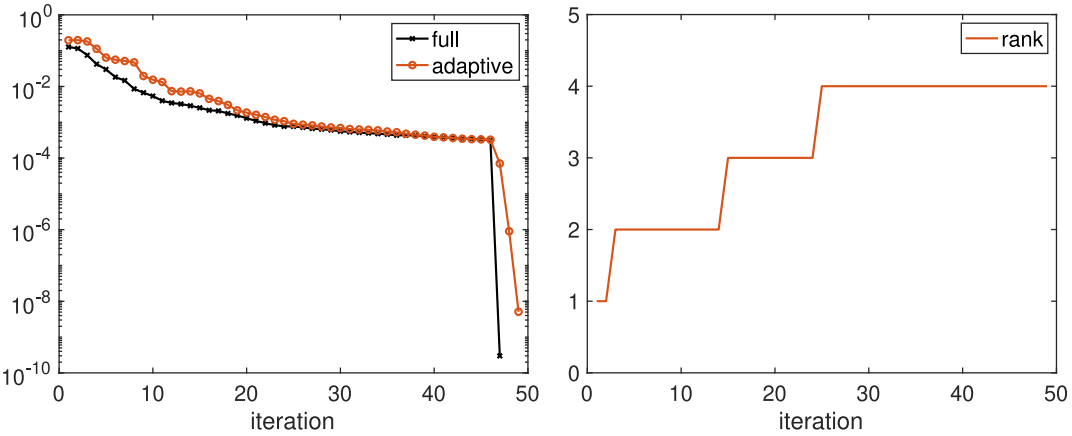


Fig. 4. Left: Residual $\|F\|_2$ as a function of the iteration number, comparing semismooth Newton with and without adaptive sketching. The methods terminated after 49 and 47 iterations, respectively. Right: Sketching target rank r as a function of the iteration number for the adaptive case (Algorithm 2).

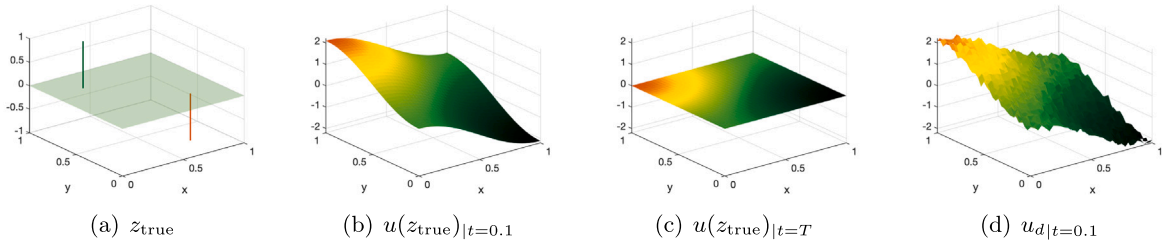
Table 1 summarizes the performance of Algorithm 2 using several fixed rank sketches, adaptive rank sketching and the full-storage method (without sketching). The adaptive algorithm was initialized with rank $r = 1$ and updated the rank as $r \leftarrow r + 1$. The notation * indicates that the semismooth Newton's method did not converge in 300 iterations. Here, the compression factor is

$$\frac{n_s n_t}{k(n_s + n_t) + s^2}.$$

Fig. 4 shows the algorithmic behavior without sketching and with adaptive sketching. The left panel shows a comparison between the residuals $\|F\|_2$ for these two cases. We observe that for a fixed rank it is possible to obtain a compression factor ~60 (for $r = 3$). With adaptive approach, we obtain a compression factor of ~50 with a final rank $r = 4$. The adaptive approach is more practical as it can be challenging to guess the rank upfront.

General measures

We consider $n_s = 32^2$ (spatial nodes) and $n_t = 501$ (time steps), so that $n_s \cdot n_t = 513,024$, We fix $\alpha = 1$ and in this experiment, we observed that it was sufficient to take a fixed $\gamma = 900$. The coarser spatial grid is motivated by the fact that now the control consists of both positive and negative parts, which are treated separately, so that we have twice as many control unknowns.



(a) z_{true} (b) $u(z_{\text{true}})_{|t=0.1}$ (c) $u(z_{\text{true}})_{|t=T}$ (d) $u_{d\,|t=0.1}$

Fig. 5. Generation of the desired state u_{d}. From left to right: True solution $z_{\text{true}} = \delta_{(0.25,0.75)} - \delta_{(0.75,0.25)}$, the associated state $u(z_{\text{true}})$ computed using Fourier modes and evaluated at $t = 0.1$ and $t = T = 2$. Last panel shows u_{d} at $t = 0.1$, which is generated by adding normally distributed random noise to $u(z_{\text{true}})$.

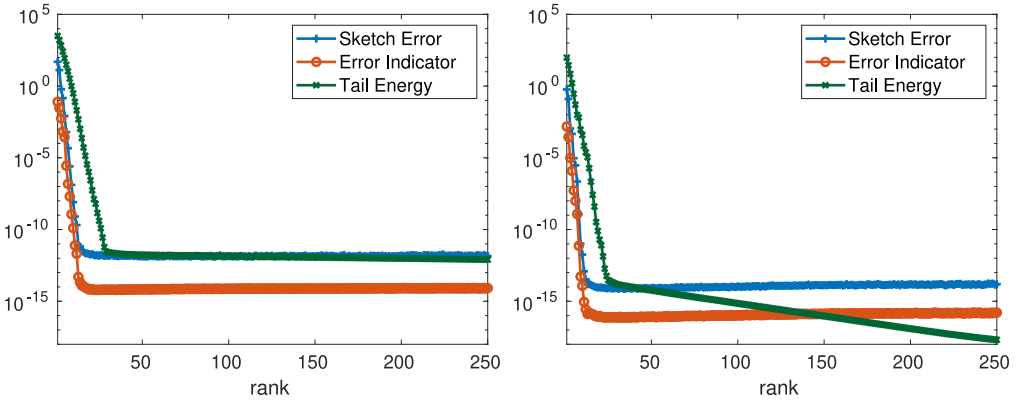
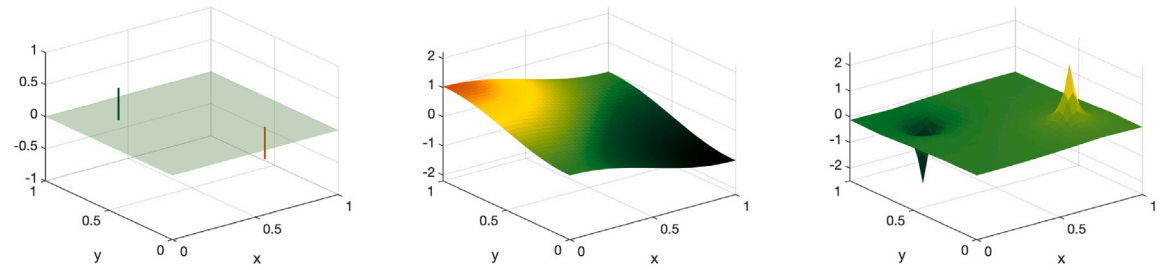


Fig. 6. The two panels display the 'sketch error' $\|\mathbf{u} - \widetilde{\mathbf{u}}\|_2$, the error indicator $\|\mathbf{c}(\widetilde{\mathbf{u}}, \mathbf{z})\|_2$, and the tail energy $\tau_{r+1}(\mathbf{u})$ for random input $(\mathbf{z}, \mu_1, \mu_2, \mu_3)$ (left) and the optimal solution $(\bar{\mathbf{z}}, \bar{\mu}_1, \bar{\mu}_2, \bar{\mu}_3)$ (right) with respect to rank r. These results corroborate the error bound (2.6). To account for randomness, the average over 10 realizations is displayed. Recall that $k = 2r + 1$ is the rank of the sketch $\widetilde{\mathbf{u}}$, therefore the sketch error curve is sometimes below the tail energy.



(a) $\bar{z}$ (b) $u(\bar{z})_{|t=0.1}$ (c) $p(\bar{z})_{|t=0}$

Fig. 7. Solution for $\alpha = 1$. From left to right: The optimal control $\bar{z}$, the optimal state $\bar{u} = u(\bar{z})$ evaluated at $t = 0.1$, and the optimal adjoint state $\bar{p} = p(\bar{z})$ evaluated at $t = 0$. The optimality conditions, (5.3a) and (5.3b) are visibly satisfied.

As before, we generate the desired state u_{d} from an initial measure control. In this case, we choose $z_{\text{true}} = \delta_{(0.25,0.75)} - \delta_{(0.75,0.25)}$, cf. Fig. 5.

Fig. 6 shows the sketch error $\|\mathbf{u} - \widetilde{\mathbf{u}}\|_2$, the error indicator $\|\mathbf{c}(\widetilde{\mathbf{u}}, \mathbf{z})\|_2$ and the tail energy of the full state $\mathbf{u}$ as functions of the sketching target rank r for random inputs $(\mathbf{z}, \mu_1, \mu_2, \mu_3)$ (left) and the optimal solution $(\bar{\mathbf{z}}, \bar{\mu}_1, \bar{\mu}_2, \bar{\mu}_3)$ (right). These results corroborate the error bound (2.6). Again, the error indicator plotted in Fig. 6 is not scaled by the constant $1/\underline{\sigma}$ as in Lemma 4.2. An exponential decay is again observed (left), which indicates that sketching will be beneficial. In fact, machine precision is achieved with relatively small r.

Fig. 7 depicts the optimal solution. The support of $\bar{z}$ coincides with the support of z_{true} (see Fig. 5), and $\|\bar{z}\|_{\mathcal{M}(\overline{\Omega})} = \alpha = 1$ holds. Furthermore, the optimal solution clearly fulfills the support subset conditions (5.3a) and (5.3b). Similar plots are obtained whether we employ Algorithm 2 with or with sketching. Consequently, we only display one of these cases.

Table 2 summarizes the performance of Algorithm 2 for several choices of fixed ranks, the adaptive sketching method with different initial ranks $r^0 \in \{1, 2, 3\}$, and the full-storage method (without sketching). The notation * indicates that the semismooth

Table 2
The performance of semismooth Newton's method for general initial measure control. Three cases are considered: Without sketching, sketching with fixed rank, and sketching with adaptive rank. The columns of the table correspond to the rank, the number of iterations, the final objective function value, the final residual $\|F\|_2$ and the compression factor. The sketched methods perform comparably to the unsketched method, but require significantly less memory.

Rank	Iterations	Objective	Residual	Compression
1*	300	3.6e+11	2.5e+07	111
2*	300	1.6e+13	7.3e+07	66
3*	300	179.2508	2.3e−01	47
4*	300	180.3544	2.9e−05	36
5	42	180.3544	1.7e−10	30
6	30	180.3544	1.1e−09	25
7	30	180.3544	8.4e−10	22
8	29	180.3544	1.0e−11	19
9	30	180.3544	1.1e−13	17
10	30	180.3544	1.2e−14	15
Adaptive ($r^0 = 1$)	53	180.3544	1.2e−09	22
Adaptive ($r^0 = 2$)	47	180.3544	1.2e−09	22
Adaptive ($r^0 = 3$)	34	180.3544	1.2e−09	22
Full	31	180.3544	5.4e−10	–

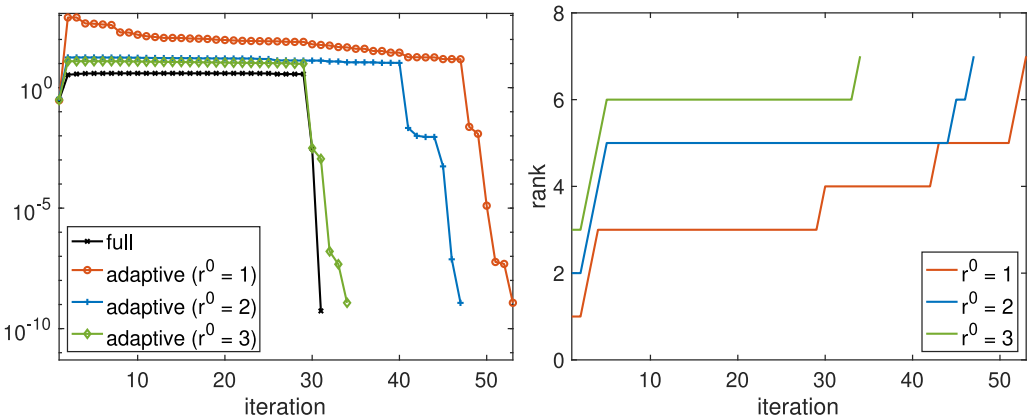


Fig. 8. Left: Residual $\|F\|_2$ as a function of the iteration, comparing semismooth Newton with and without adaptive sketching for different choices of initial target rank r^0. The methods terminated after 31 (full), 53 ($r^0 = 1$), 47 ($r^0 = 2$), and 34 ($r^0 = 3$) Newton steps, respectively. Right: Sketching target rank r as a function of the iteration for the adaptive cases (Algorithm 2). Notice that the norm of the residual goes slightly up in the first iteration in the left panel. This is because, the line search failed in the first iteration (after 10 reductions), but after this iteration, we observe a monotonic behavior in all cases.

Newton's method did not converge in 300 iterations. Fig. 8 demonstrates the algorithmic performance with and without sketching. The left panel shows a comparison between the residuals $\|F\|_2$ for the four portrayed cases. The adaptive algorithm produced a final rank of $r = 7$ for each initial rank $r^0 \in \{1, 2, 3\}$, yielding a compression factor of ~ 22. Initialization with a slightly larger target rank led to a significant reduction of generalized Newton iterations.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

We are grateful to Dr. Konstantin Pieper for pointing out an incorrect statement in Lemma 5 in the original draft of the paper.

Appendix A. Proof of Theorem 4.7

Proof. The definition of $w(z;v)$ from (4.4) yields $e_u(u(z),z)w(z;v) + e_z(u(z),z)v = 0$ and therefore we have for any $z \in Z_0$, $v \in Z$, and $w \in U$ that

$$\zeta(u(z),z) := e_u(u(z),z)w + e_z(u(z),z)v = e_u(u(z),z)(w - w(z;v)). \tag{A.1}$$

From (A.1) and invoking Assumption 4.1(b), we immediately obtain that

$$\underline{\sigma}\|w - w(z;v)\|_U \le \|\zeta(u(z),z)\|_W \le \overline{\sigma}\|w - w(z;v)\|_U. \tag{A.2}$$

Furthermore, we can estimate

$$\|\zeta(u(z),z)\|_W \le \|\zeta(u,z)\|_W + \|\zeta(u(z),z) - \zeta(u,z)\|_W.$$

Employing the Lipschitz-continuity of e_u and e_z from Assumption 4.1(c), we further estimate the second summand on the right-hand side

$$\begin{aligned}
&\|\zeta(u(z),z) - \zeta(u,z)\|_W \\
&\quad \le \|e_u(u(z),z) - e_u(u,z)\|_{\mathscr{L}(U,W)}\|w\|_U + \|e_z(u(z),z) - e_z(u,z)\|_{\mathscr{L}(Z,W)}\|v\|_Z \\
&\quad \lesssim (1 + \|v\|_Z)\,\|u - u(z)\|_U \\
&\quad \overset{(4.2a)}{\lesssim} \frac{(1+\|v\|_Z)}{\overline{\sigma}}\|e(u,z)\|_W
\end{aligned}$$

where the hidden constant incorporates the Lipschitz constants for e_u and e_z and norm bound on $\|w\|_U$. Inserting the above inequalities into (A.2), we arrive at

$$\|w - w(z;v)\|_U \lesssim \frac{1}{\underline{\sigma}}\|e_u(u,z)w + e_z(u,z)v\|_W + \frac{(1+\|v\|_Z)}{\underline{\sigma}\overline{\sigma}}\|e(u,z)\|_W,$$

which proves (4.5a). The estimator for q works in a similar manner. By definition of $q(z;v)$ from (4.4) we have that $e_u(u(z),z)^* q(z;v) - \mathcal{L}_{uu}(u(z),z,p(z))w(z;v) - \mathcal{L}_{uz}(u(z),z,p(z))v = 0$. Consequently, we obtain that

$$\begin{aligned}
\zeta_2(u(z),z) &:= e_u(u(z),z)^* q - \mathcal{L}_{uu}(u(z),z,p(z))w(z;v) - \mathcal{L}_{uz}(u(z),z,p(z))v \\
&= e_u(u(z),z)^*(q - q(z;v)).
\end{aligned}$$

By Assumption 4.1(b), we deduce that

$$\underline{\sigma}\|q - q(z;v)\|_{W^*} \le \|\zeta_2(u(z),z)\|_{U^*} \le \overline{\sigma}\|q - q(z;v)\|_{W^*}. \tag{A.3}$$

Next, we estimate

$$\begin{aligned}
\|\zeta_2(u(z),z)\|_{U^*} &\le \|\zeta_2(u,z)\|_{U^*} + \|e_u(u(z),z)^* q - e_u(u,z)^* q\|_{U^*} \\
&\quad + \|\mathcal{L}_{uu}(u(z),z,p(z))w(z;v) - \mathcal{L}_{uu}(u,z,p)w\|_{U^*} + \|\mathcal{L}_{uz}(u(z),z,p(z))v - \mathcal{L}_{uz}(u,z,p)v\|_{U^*} \\
&= \text{(I)} + \text{(II)} + \text{(III)} + \text{(IV)}.
\end{aligned} \tag{A.4}$$

By Lipschitz continuity of e_u, we have

$$\text{(II)} = \|e_u(u(z),z)^* q - e_u(u,z)^* q\|_{U^*} \lesssim \|u - u(z)\|_U.$$

Also, it holds

$$\begin{aligned}
\text{(III)} &\le \|\mathcal{L}_{uu}(u(z),z,p(z))\|_{\mathscr{L}(U,U^*)}\|w(z;v) - w\|_U + \|\mathcal{L}_{uu}(u(z),z,p(z)) - \mathcal{L}_{uu}(u,z,p)\|_{\mathscr{L}(U,U^*)}\|w\|_U \\
&\lesssim \|w(z;v) - w\|_U + \|\mathcal{L}_{uu}(u(z),z,p(z)) - \mathcal{L}_{uu}(u,z,p(z)) + \mathcal{L}_{uu}(u,z,p(z)) - \mathcal{L}_{uu}(u,z,p)\|_{\mathscr{L}(U,U^*)} \\
&\lesssim \|w(z;v) - w\|_U + \|u - u(z)\|_U + \|p - p(z)\|_{W^*},
\end{aligned}$$

where we used the boundedness of $\|\mathcal{L}_{uu}(u(z),z,p(z))\|_{\mathscr{L}(U,U^*)}$ and $\|w\|_U$ in the second inequality and the Lipschitz continuity of $\mathcal{L}_{uu}$ in its first and third argument in the third inequality. Similarly, we exploit the Lipschitz continuity of $\mathcal{L}_{uz}$ in its first and third argument to deduce

$$\text{(IV)} = \|\mathcal{L}_{uz}(u(z),z,p(z))v - \mathcal{L}_{uz}(u,z,p)v\|_{U^*} \lesssim \|v\|_Z\left(\|u - u(z)\|_U + \|p - p(z)\|_{W^*}\right)$$

Collecting the estimates for (II), (III), and (IV), and substituting in (A.4), we obtain

$$\begin{aligned}
\|\zeta_2(u(z),z)\|_{U^*} &\lesssim \|e_u(u,z)^* q - \mathcal{L}_{uu}(u,z,p)w - \mathcal{L}_{uz}(u,z,p)v\|_{U^*} \\
&\quad + (1 + \|v\|_Z)\|u - u(z)\|_U + (1 + \|v\|_Z)\|p - p(z)\|_{W^*} + \|w - w(z;v)\|_U.
\end{aligned}$$

Combining this inequality with (A.3) and then using the available estimates for the state u (4.2a), the adjoint p (4.2b), and w (4.5a) we deduce the inequality (4.5b).

Finally, we will prove the estimate related to the Hessian-vector product (4.5c). From (4.4), we recall that $\hat{J}''(z)v = -e_z(u(z),z)^* q(z;v) + \mathcal{L}_{zu}(u(z),z,p(z))w(z;v) + \mathcal{L}_{zz}(u(z),z,p(z))v$. This allows us to estimate

$$\begin{aligned}
&\| - e_z(u,z)^* q + \mathcal{L}_{zu}(u,z,p)w + \mathcal{L}_{zz}(u,z,p)v - \hat{J}''(z)v\|_{Z^*} \\
&\quad \le \|e_z(u,z)^* q - e_z(u(z),z)^* q(z;v)\|_{Z^*} + \|\mathcal{L}_{zu}(u,z,p)w - \mathcal{L}_{zu}(u(z),z,p(z))w(z;v)\|_{Z^*} \\
&\qquad + \|\mathcal{L}_{zz}(u,z,p)v - \mathcal{L}_{zz}(u(z),z,p(z))v\|_{Z^*} \\
&\quad = \text{(i)} + \text{(ii)} + \text{(iii)}.
\end{aligned} \tag{A.5}$$

Next, we estimate (i), (ii), and (iii). It follows that

$$\begin{aligned}
\text{(i)} &\le \|e_z(u,z)^* q - e_z(u,z)^* q(z;v)\|_{Z^*} + \|e_z(u,z)^* q(z;v) - e_z(u(z),z)^* q(z;v)\|_{Z^*} \\
&\lesssim \|q - q(z;v)\|_{W^*} + \|u - u(z)\|_U,
\end{aligned}$$

where we used the boundedness of $\|e_z(u,z)^*\|_{\mathscr{L}(W^*,Z^*)}$ and $\|q(z;v)\|_{W^*}$, and the Lipschitz-continuity of e_z in its first argument. We proceed and estimate

$$\begin{aligned}
\text{(ii)} &\le \|\mathcal{L}_{zu}(u,z,p)w - \mathcal{L}_{zu}(u,z,p)w(z;v)\|_{Z^*} \\
&\qquad + \|\mathcal{L}_{zu}(u,z,p)w(z;v) - \mathcal{L}_{zu}(u(z),z,p)w(z;v)\|_{Z^*} \\
&\qquad + \|\mathcal{L}_{zu}(u(z),z,p)w(z;v) - \mathcal{L}_{zu}(u(z),z,p(z))w(z;v)\|_{Z^*} \\
&\lesssim \|w - w(z;v)\|_U + \|u - u(z)\|_U + \|p - p(z)\|_{W^*},
\end{aligned}$$

where we used the boundedness of $\|\mathcal{L}_{zu}(u,z,p)\|_{\mathscr{L}(U,Z^*)}$ and $\|w(z;v)\|_U$, and the Lipschitz continuity of $\mathcal{L}_{zu}$ in its first and third argument. Similarly, we can use Lipschitz continuity of $\mathcal{L}_{zz}$ in its first and third argument to see

$$\text{(iii)} \lesssim \|v\|_Z \left(\|u - u(z)\|_U + \|p - p(z)\|_{W^*}\right).$$

Substituting the estimates of (i), (ii), and (iii) in (A.5), we obtain that

$$\begin{aligned}
&\| - e_z(u,z)^* q + \mathcal{L}_{zu}(u,z,p)w + \mathcal{L}_{zz}(u,z,p)v - \hat{J}''(z)v\|_{Z^*} \\
&\quad \lesssim (1 + \|v\|_Z)\|u - u(z)\|_U + (1 + \|v\|_Z)\|p - p(z)\|_{W^*} + \|w - w(z;v)\|_U + \|q - q(z;v)\|_{W^*}.
\end{aligned}$$

We can now insert the known estimates (4.2a), (4.2b), (4.5a), and (4.5b) to assemble (4.5c). This concludes the proof. □

Appendix B. Proof of Theorem 3.1

Proof. We proceed in two steps. In the first step we show that the complementarity conditions (3.3c) imply (3.4). In the second step, we establish the reverse implication.

Step I: From the optimality of the projection, we have

$$0 \le \left(\text{proj}_{K^-}(\mu + \kappa g(z)) - (\mu + \kappa g(z)),\, k - \text{proj}_{K^-}(\mu + \kappa g(z))\right)_{\mathcal{G}} \quad \forall\, k \in K^-. \tag{B.1}$$

Since $\mu \in K^-$ from (3.3c), we can replace k by μ in (B.1) to arrive at

$$\begin{aligned}
0 &\le \left(\text{proj}_{K^-}(\mu + \kappa g(z)) - (\mu + \kappa g(z)),\, \mu - \text{proj}_{K^-}(\mu + \kappa g(z))\right)_{\mathcal{G}} \\
&= -\|\mu - \text{proj}_{K^-}(\mu + \kappa g(z))\|_{\mathcal{G}}^2 - \kappa \left(g(z),\, \mu - \text{proj}_{K^-}(\mu + \kappa g(z))\right)_{\mathcal{G}} \\
&= -\|\mu - \text{proj}_{K^-}(\mu + \kappa g(z))\|_{\mathcal{G}}^2 + \kappa \left(g(z),\, \text{proj}_{K^-}(\mu + \kappa g(z))\right)_{\mathcal{G}} = \text{I} + \text{II}
\end{aligned} \tag{B.2}$$

where in the last equation we used the fact that $(g(z),\mu)_G = 0$. Combining the facts that $\text{proj}_{K^-}(\mu + \kappa g(z)) \in K^-$ and $g(z) \in K$ with the definition of polar cone K^-, we obtain that the second term in the final equation of (B.2) fulfills $\text{II} \le 0$. Thus from (B.2), we arrive at (3.4).

Step II: Since $\mu = \text{proj}_{K^-}(\mu + \kappa g(z))$, we have that $\mu \in K^-$. Next, we show that $g(z) \in K$. Using Moreau's decomposition, we have that

$$\begin{aligned}
\mu + \kappa g(z) &= \text{proj}_{K^-}(\mu + \kappa g(z)) + \text{proj}_K(\mu + \kappa g(z)) \\
&= \mu + \text{proj}_K(\mu + \kappa g(z)).
\end{aligned}$$

Combining this with the fact that K is a cone yields

$$g(z) = \kappa^{-1}\text{proj}_K(\mu + \kappa g(z)) \in K.$$

It remains to show that $(g(z),\mu)_{\mathcal{G}} = 0$. As in Step I:, we use (B.1). Since μ satisfies (3.4), we can substitute it into (B.1)to obtain we obtain

$$0 \le (\mu - (\mu + \kappa g(z)),\, k - \mu)_G = -\kappa\,(g(z),\, k - \mu)_G \quad \forall\, k \in K^-,$$

which leads to

$$(g(z), k-\mu)_G \le 0 \quad \forall\, k \in K^-.$$

Since $\mu \in K^-$ and K^- is a cone, we can select $k = b\mu \in K^-$ for any $b > 0$ to obtain

$$(b-1)(g(z), \mu)_{\mathcal{G}} \le 0. \tag{B.3}$$

The left-hand side in (B.3) is negative if $b > 1$ and positive if $b \in (0, 1)$, allowing use to conclude that $(g(z), \mu)_{\mathcal{G}} = 0$. □

Appendix C. Algorithms

The following algorithms describe how to compute state sketching and error in the state equation due to sketching. Here by c_i we indicate the ith row of the linear state equation, cf. (1.1).

Algorithm 3 Sketching of the state with initial control

Require: The control iterate $\mathbf{z} \in \mathbb{R}^{n_s}$ and the target rank r.

Set $k = 2r+1, s = 2k+1$.
Initialize random matrices $\boldsymbol{\Upsilon} \in \mathbb{R}^{k\times n_s}, \boldsymbol{\Omega} \in \mathbb{R}^{k\times n_t}, \boldsymbol{\Phi} \in \mathbb{R}^{s\times n_s}, \boldsymbol{\Psi} \in \mathbb{R}^{s\times n_t}$.
Determine $\mathbf{u}_{\text{curr}}$ from initial control $\mathbf{z}$.
for $i = 1 : n_t$ **do**
  Solve $c_i(\mathbf{u}_{\text{curr}}, \mathbf{u}_{\text{next}}) = 0$ for $\mathbf{u}_{\text{next}}$.
  Update sketching matrices $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ with the i-th column of the state $\mathbf{u}_{\text{next}}$.
  Set $\mathbf{u}_{\text{curr}} = \mathbf{u}_{\text{next}}$.
end for
Form skinny matrices $\mathbf{Q}$ and $\mathbf{W}$.

Algorithm 4 Calculation of the sketching error in the state equation

Require: The control iterate $\mathbf{z} \in \mathbb{R}^{n_s}$ and the skinny matrices $\mathbf{Q} \in \mathbb{R}^{n_s\times k}, \mathbf{W} \in \mathbb{R}^{k\times n_t}$.

Determine $\widetilde{\mathbf{u}}_{\text{curr}}$ from initial control $\mathbf{z}$.
for $i = 1 : n_t$ **do**
  Reconstruct $\widetilde{\mathbf{u}}_{\text{next}} = (\mathbf{QW})(:, i)$.
  Compute the error $err_i = c_i(\widetilde{\mathbf{u}}_{\text{curr}}, \widetilde{\mathbf{u}}_{\text{next}})$.
  Set $\widetilde{\mathbf{u}}_{\text{curr}} = \widetilde{\mathbf{u}}_{\text{next}}$.
end for
Compute $\|c(\widetilde{\mathbf{U}}, \mathbf{z})\| = \sum_{i=1}^{n_t} err_i$.

References

[1] R. Löhner, H. Antil, J.M. Gimenez, S. Idelsohn, E. Oñate, A deterministic pathogen transmission model based on high-fidelity physics, Comput. Methods Appl. Mech. Engrg. 401 (2022) Paper No. 114929.

[2] R. Löhner, H. Antil, A. Srinivasan, S. Idelsohn, E. Oñate, High-fidelity simulation of pathogen propagation, transmission and mitigation in the built environment, Arch. Comput. Methods Eng. (2021) 1–26.

[3] D. Dentcheva, W. Römisch, Optimal power generation under uncertainty via stochastic programming, in: Stochastic Programming Methods and Technical Applications, Neubiberg/MUnich, 1996, in: Lecture Notes in Econom. and Math. Systems, vol. 458, Springer, Berlin, 1998, pp. 22–56.

[4] K. Harada, T. Matsuda, J. Bonevich, M. Igarashi, S. Kondo, G. Pozzi, U. Kawabe, A. Tonomura, Real-time observation of vortex lattices in a superconductor by electron microscopy, Nature 360 (6399) (1992) 51–53.

[5] H. Antil, R.H. Nochetto, P. Venegas, Controlling the Kelvin force: Basic strategies and applications to magnetic drug targeting, Optim. Eng. 19 (3) (2018) 559–589.

[6] H. Antil, D.P. Kouri, M.-D. Lacasse, D. Ridzal (Eds.), Frontiers in PDE-Constrained Optimization, The IMA Volumes in Mathematics and its Applications, vol. 163, Springer, New York, 2018, p. x+434, Papers based on the workshop held at the Institute for Mathematics and its Applications, Minneapolis, MN, June 6–10, 2016.

[7] H. Antil, M. Heinkenschloss, R.H.W. Hoppe, Domain decomposition and balanced truncation model reduction for shape optimization of the Stokes system, Optim. Methods Softw. 26 (4–5) (2011) 643–669.

[8] H. Antil, M. Heinkenschloss, R.H.W. Hoppe, D.C. Sorensen, Domain decomposition and model reduction for the numerical solution of PDE constrained optimization problems with localized optimization variables, Comput. Vis. Sci. 13 (6) (2010) 249–264.

[9] A.C. Antoulas, C.A. Beattie, S. Güğercin, Interpolatory Methods for Model Reduction, SIAM, 2020.

[10] J.S. Hesthaven, G. Rozza, B. Stamm, Certified Reduced Basis Methods for Parametrized Partial Differential Equations, in: SpringerBriefs in Mathematics, Springer, Cham; BCAM Basque Center for Applied Mathematics, Bilbao, 2016, p. xiii+131, BCAM SpringerBriefs.

[11] K. Ito, S.S. Ravindran, Reduced basis method for optimal control of unsteady viscous flows, Int. J. Comput. Fluid Dyn. 15 (2) (2001) 97–113.

[12] A. Quarteroni, A. Manzoni, F. Negri, Reduced Basis Methods for Partial Differential Equations, in: Unitext, vol. 92, Springer, Cham, 2016, p. xi+296, An introduction, La Matematica per il 3+2.

[13] M. Fahl, E.W. Sachs, Reduced order modelling approaches to PDE-constrained optimization based on proper orthogonal decomposition, in: Large-Scale PDE-Constrained Optimization, Santa Fe, NM, 2001, in: Lect. Notes Comput. Sci. Eng., vol. 30, Springer, Berlin, 2003, pp. 268–280.

[14] M.J. Zahr, K.T. Carlberg, D.P. Kouri, An efficient, globally convergent method for optimization under uncertainty using adaptive model reduction and sparse grids, SIAM/ASA J. Uncertain. Quantif. 7 (3) (2019) 877–912.

[15] H. Antil, M. Heinkenschloss, D.C. Sorensen, Application of the discrete empirical interpolation method to reduced order modeling of nonlinear and parametric systems, in: Reduced Order Methods for Modeling and Computational Reduction, in: MS&A. Model. Simul. Appl., vol. 9, Springer, Cham, 2014, pp. 101–136.
[16] M. Barrault, Y. Maday, N.C. Nguyen, A.T. Patera, An 'empirical interpolation' method: Application to efficient reduced-basis discretization of partial differential equations, C. R. Math. Acad. Sci. Paris 339 (9) (2004) 667–672.
[17] S. Chaturantabut, D.C. Sorensen, Nonlinear model reduction via discrete empirical interpolation, SIAM J. Sci. Comput. 32 (5) (2010) 2737–2764.
[18] A. Griewank, A. Walther, Algorithm 799: revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation, ACM Trans. Math. Softw. 26 (1) (2000) 19–45.
[19] P. Stumm, A. Walther, New algorithms for optimal online checkpointing, SIAM J. Sci. Comput. 32 (2) (2010) 836–854.
[20] Q. Wang, P. Moin, G. Iaccarino, Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation, SIAM J. Sci. Comput. 31 (4) (2009) 2549–2567.
[21] R. Muthukumar, D.P. Kouri, M. Udell, Randomized sketching algorithms for low-memory dynamic optimization, SIAM J. Optim. 31 (2) (2021) 1242–1275.
[22] J.A. Tropp, A. Yurtsever, M. Udell, V. Cevher, Fixed-rank approximation of a positive-semidefinite matrix from streaming data, Adv. Neural Inf. Process. Syst. 30 (2017).
[23] J.A. Tropp, A. Yurtsever, M. Udell, V. Cevher, Practical sketching algorithms for low-rank matrix approximation, SIAM J. Matrix Anal. Appl. 38 (4) (2017) 1454–1485.
[24] J.A. Tropp, A. Yurtsever, M. Udell, V. Cevher, Streaming low-rank matrix approximation with an application to scientific simulation, SIAM J. Sci. Comput. 41 (4) (2019) A2430–A2463.
[25] M. Ulbrich, Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces, in: MOS-SIAM Series on Optimization, vol. 11, Society for Industrial and Applied Mathematics (SIAM), Mathematical Optimization Society, Philadelphia, PA, Philadelphia, PA, 2011, p. xiv+308.
[26] M. Hinze, R. Pinnau, M. Ulbrich, S. Ulbrich, Optimization with PDE Constraints, in: Mathematical Modelling: Theory and Applications, vol. 23, Springer, New York, 2009, p. xii+270.
[27] R.J. Baraldi, D.P. Kouri, A proximal trust-region method for nonsmooth optimization with inexact function and gradient evaluations, Math. Program. (2022) 1–40.
[28] D.P. Kouri, D. Ridzal, Inexact trust-region methods for PDE-constrained optimization, Front. PDE-Constrain. Optim. (2018) 83–121.
[29] J.E. Dennis, J.J. Moré, A characterization of superlinear convergence and its application to quasi-Newton methods, Math. Comput. 28 (126) (1974) 549–560.
[30] A.L. Dontchev, Generalizations of the Dennis–Moré theorem, SIAM J. Optim. 22 (3) (2012) 821–830.
[31] J. Nocedal, S.J. Wright, Numerical Optimization, in: Springer Series in Operations Research, Springer-Verlag, New York, 1999, p. xxii+636.
[32] M. Ulbrich, S. Ulbrich, Nichtlineare Optimierung, Springer-Verlag, 2012.
[33] C. Bennewitz, Approximation numbers=singular values, J. Comput. Appl. Math. 208 (1) (2007) 102–110, Special Issue: 65th birthday of Prof. Desmond Evans.
[34] H. Antil, D. Leykekhman, A brief introduction to PDE-constrained optimization, in: Frontiers in PDE-Constrained Optimization, in: IMA Math. Appl., vol. 163, Springer, New York, 2018, pp. 3–40.
[35] S. Götschel, M. Weiser, Lossy compression for PDE-constrained optimization: Adaptive error control, Comput. Optim. Appl. 62 (1) (2015) 131–155.
[36] Y. Saad, M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.
[37] W. Rudin, Real and Complex Analysis, McGraw-Hill Book Co., 1970.
[38] E. Casas, K. Kunisch, Using sparse control methods to identify sources in linear diffusion-convection equations, Inverse Problems 35 (11) (2019) 114002, 17.
[39] M. Hinze, A variational discretization concept in control constrained optimization: The linear-quadratic case, Comput. Optim. Appl. 30 (1) (2005) 45–61.
[40] E. Herberg, M. Hinze, Variational discretization approach applied to an optimal control problem with bounded measure controls, in: Optimization and Control for Partial Differential Equations—Uncertainty Quantification, Open and Closed-Loop Control, and Shape Optimization, in: Radon Ser. Comput. Appl. Math., 29 (2022) 113–135.
[41] E. Herberg, M. Hinze, H. Schumacher, Maximal discrete sparsity in parabolic optimal control with measures, Math. Control Relat. Fields 10 (4) (2020) 735–759.
[42] E. Herberg, Sparse discretization of sparse control problems with measures (Ph.D. thesis), Universität Koblenz-Landau, Universitätsbibliothek, 2021, p. viii, 106.