Optimization and Engineering manuscript No.
(will be inserted by the editor)

A hybrid semismooth quasi-Newton method for nonsmooth optimal control with PDEs

Florian Mannel · Armin Rund

Received: date / Accepted: date

Abstract We propose a semismooth Newton-type method for nonsmooth optimal control problems. Its particular feature is the combination of a quasi-Newton method with a semismooth Newton method. This reduces the computational costs in comparison to semismooth Newton methods while maintaining local superlinear convergence. The method applies to Hilbert space problems whose objective is the sum of a smooth function, a regularization term, and a nonsmooth convex function.

In the theoretical part of this work we establish the local superlinear convergence of the method in an infinite-dimensional setting and discuss its application to sparse optimal control of the heat equation subject to box constraints. We verify that the assumptions for local superlinear convergence are satisfied in this application and we prove that convergence can take place in stronger norms than that of the Hilbert space if initial error and problem data permit.

In the numerical part we provide a thorough study of the hybrid approach on two optimal control problems, including an engineering problem from magnetic resonance imaging that involves bilinear control of the Bloch equations. We use this problem to demonstrate that the new method is capable of solving nonconvex, nonsmooth large-scale real-world problems. Among others, the study addresses mesh independence, globalization techniques, and limited-memory methods. We observe throughout that algorithms based on the hybrid methodology are several times faster in runtime than their semismooth Newton counterparts.

Keywords Semismooth Newton methods · quasi-Newton methods · superlinear convergence · nonsmooth optimal control · Bloch equations

Mathematics Subject Classification (2010) 49M05 · 49M15 · 49K20 · 49-04 · 90-04 · 90C06 · 90C30 · 90C53 · 90C56 · 90C90

F. Mannel / A. Rund
University of Graz, Heinrichstr. 36, 8010 Graz, Austria
E-mail: florian.mannel@uni-graz.at / E-mail: armin.rund@uni-graz.at

1 Introduction

In this paper we present a novel algorithm to solve nonsmooth optimization problems of the form

$$\min_{u\in U} \hat{f}(u) + \frac{\gamma}{2}\,\|u\|_U^2 + \varphi(u) \tag{P}$$

and apply it to optimal control problems that are special instances of (P). Here, U is a Hilbert space, $\gamma > 0$, $\varphi : U \longrightarrow \mathbb{R} \cup \{+\infty\}$ is convex but possibly nonsmooth, and $\hat{f} : U \longrightarrow \mathbb{R}$ is smooth but possibly nonconvex. The precise problem setting is given in Section 3. A prototypical example from PDE-constrained optimal control is

$$\min_{u\in L^2(\Omega)} \frac{1}{2}\,\|y(u) - y_d\|_{L^2(\Omega)}^2 + \frac{\alpha}{2}\,\|u\|_{L^2(\Omega)}^2 + \beta\,\|u\|_{L^1(\Omega)} \qquad \text{s.t.} \qquad a \le u \le b \quad \text{a.e. in } \Omega,$$

where $a < 0 < b$ and $\alpha, \beta > 0$ are real numbers, Ω is a bounded Lipschitz domain, $y_d \in L^2(\Omega)$, and $y = y(u)$ is for $u \in L^2(\Omega)$ the solution of the semilinear equation $-\Delta y + y^3 = u$ with appropriate boundary conditions. As $\hat{f}(u) = \frac{1}{2}\|y(u) - y_d\|_{L^2(\Omega)}^2$ in this instance of (P), the evaluation of $\hat{f}$ and its derivatives requires PDE solves.

The new algorithm is a semismooth Newton-type method that exploits the presence of the smooth term $\nabla\hat{f}$ in the optimality conditions of (P) by applying a quasi-Newton method to it. Consequently, the operator $\nabla^2\hat{f}(u^k)$ that appears in semismooth Newton methods is replaced by a quasi-Newton approximation B_k. In PDE-constrained optimal control problems this lowers the runtime significantly because it omits the PDE solves that occur in the evaluation of Hessian-vector products $\nabla^2\hat{f}(u^k)d$ while maintaining superlinear convergence. Note that a direct application of quasi-Newton methods to semismooth equations cannot ensure superlinear convergence. For instance, Broyden's method on a piecewise affine (hence semismooth) equation in $\mathbb{R}$ may converge only r-linearly, cf. [9, Introduction]. We present the hybrid method and its convergence properties for problems of the form (P) in Section 3. Its application to a model problem from PDE-constrained optimal control, the time-dependent sparse optimal control of the heat equation subject to box constraints, constitutes Section 4 and concludes the theoretical part of this paper.

The remainder of the paper is devoted to numerics. We devise various numerical realizations of the hybrid method and compare them as part of an extensive numerical study. This study addresses many theoretical and practical aspects of the hybrid approach: We verify experimentally the superlinear convergence with respect to different norms, we investigate mesh independence properties, we compare different globalization strategies including a trust-region method, and we examine different quasi-Newton updates (Broyden, SR1, BFGS).
The numerical study is based on two optimal control problems, starting with the time-dependent box-constrained sparse optimal control of the linear heat equation from the theoretical part. This model problem allows us to display very clearly the convergence properties of the hybrid method, e.g., its local superlinear convergence. The second problem deals with the design of radio-frequency pulses for magnetic resonance imaging, a topic from medical engineering. A realistic modeling yields a nonsmooth and nonconvex optimal control problem that serves as a benchmark for the performance of the hybrid approach on real-world applications. The numerical

results underline that the hybrid approach can be competitive on such problems. Indeed, a previous version of the presented trust-region method formed the kernel of the code [21, 22] that won the ISMRM challenge on radio-frequency pulse design in magnetic resonance imaging [10]. Here we provide an improved successor.

Since quasi-Newton methods involve the Hilbert space structure of U in an essential way, it may be surprising that the numerical results for both problems clearly indicate convergence of the control u with respect to stronger norms than that of U. In Section 4 we establish rigorous theoretical results that explain this behavior for the control of the heat equation. It is related to the regularity of the problem and the quality of the initial approximation.

Although quasi-Newton methods have been applied to PDE-constrained optimal control problems, for instance in [5, Chapter 4], [11], and [27, Chapter 11], there are rather few infinite-dimensional convergence results available for algorithms that incorporate quasi-Newton methods and can handle nonsmoothness. We are aware of [1, 9, 16, 23], but none of these yield superlinear convergence for (P).

State-of-the-art methods that can solve optimal control problems of the form (P) are semismooth Newton methods, cf. [8, 12, 13, 27]. In the numerical study we therefore consistently compare the hybrid approach to semismooth Newton methods.

This paper is organized as follows. Section 2 specifies some notions that are important for this work, e.g., semismoothness. In Section 3 we provide the problem under consideration in full detail, introduce the hybrid method, and establish convergence results for it. Section 4 discusses the application of the method to sparse optimal control of the linear heat equation with box constraints. In Section 5 we comment on implementation issues. Section 6 contains the numerical study and in Section 7 we draw conclusions from this work. In Appendix A we provide the algorithm that we found most effective in the numerical studies—a matrix-free limited-memory truncated trust-region variant of the hybrid method.

2 Preliminaries

In this short section we fix the notation, recall the notion of proximal maps, and specify which concept of semismoothness we use.

We set $\mathbb{N} := \{1, 2, 3, \dots\}$. All linear spaces are real linear spaces. Let X and Y be Banach spaces. We denote $\mathbb{B}_\delta(\bar{x}) := \{x \in X : \|x - \bar{x}\|_X < \delta\}$ for $\delta > 0$ and $\bar{x} \in X$. Moreover, $\mathcal{L}(X, Y)$ represents the space of bounded linear maps from X to Y. If X can be continuously embedded into Y, this is indicated by $X \hookrightarrow Y$. In the Hilbert space U we write $(v, w)_U$ for the scalar product of $v, w \in U$ and $(v, \cdot)_U$ for the linear operator $w \mapsto (v, w)_U$ from U to $\mathbb{R}$. Furthermore, we recall the definition and elementary properties of the proximal map.

Definition 1 Let U be a Hilbert space and let $\gamma > 0$. Let $\varphi : U \to \mathbb{R} \cup \{+\infty\}$ be a proper, convex, and lower semicontinuous function and denote its effective domain by $C := \{u \in U : \varphi(u) < +\infty\}$. The *proximal mapping of* $\varphi_\gamma := \frac{\varphi}{\gamma}$ is given by

$$\operatorname{Prox}_{\varphi_\gamma} : U \to U, \quad \operatorname{Prox}_{\varphi_\gamma}(u) := \operatorname{argmin}_{\tilde{u} \in C} \left[\frac{1}{2} \|\tilde{u} - u\|_U^2 + \varphi_\gamma(\tilde{u}) \right].$$

It is easy to see that $\mathrm{Prox}_{\varphi_\gamma}$ is single-valued, has image C, and satisfies the relation

$$\hat{u} = \mathrm{Prox}_{\varphi_\gamma}(u) \qquad \Longleftrightarrow \qquad \gamma(u - \hat{u}) \in \partial\varphi(\hat{u}) \tag{1}$$

for $u, \hat{u} \in U$ and arbitrary $\gamma > 0$, where $\partial\varphi$ denotes the convex subdifferential of φ. If φ is the characteristic function of a closed convex set, then $\mathrm{Prox}_{\varphi_\gamma}$ is the projection onto that set. More on proximal mappings in Hilbert spaces can be found in [3, Section 24], for instance.

We will use the rather general notion of semismoothness from [27, Definition 3.1] that includes, for instance, Newton differentiability, cf. [13, Definition 8.10].

Definition 2 Let X, Y be Banach spaces and let $\bar{x} \in X$. Let $G : X \longrightarrow Y$ be continuous in an open neighborhood of $\bar{x}$. Moreover, let $\partial G : X \rightrightarrows \mathcal{L}(X, Y)$ satisfy $\partial G(x) \neq \emptyset$ for all $x \in X$. We say that G is *semismooth at $\bar{x}$ with respect to ∂G* iff there holds

$$\sup_{M \in \partial G(\bar{x}+h)} \|G(\bar{x} + h) - G(\bar{x}) - Mh\|_Y = o(\|h\|_X) \qquad \text{for } \|h\|_X \longrightarrow 0.$$

The set-valued mapping $\partial G : X \rightrightarrows \mathcal{L}(X, Y)$ is called a *generalized derivative of G*. For $x \in X$ every $M \in \partial G(x)$ is called a *generalized differential of G at x*.

3 Problem setting, algorithm, and convergence results

In this section we introduce the problem class in full detail and present the hybrid method to solve it. We provide its convergence properties and recall a result concerning local uniform invertibility of generalized differentials.

3.1 Problem setting and algorithm

Throughout this work we consider optimization problems of the form

$$\min_{u \in U} \underbrace{\hat{f}(u) + \frac{\gamma}{2}\|u\|_U^2}_{=:f(u)} + \varphi(u), \tag{P}$$

the details of which are contained in the following assumption.

Assumption 1 *Let the following conditions be satisfied.*

1) *U is a Hilbert space.*
2) *(P) has a local solution, denoted $\bar{u} \in U$.*
3) *The function $\varphi : U \longrightarrow \mathbb{R} \cup \{+\infty\}$ is proper, convex, and lower semicontinuous.*
4) *The function $\hat{f} : U \longrightarrow \mathbb{R}$ is continuously differentiable.*
5) *There is a Banach space $Q \hookrightarrow U$ such that*

$$\nabla\hat{f}(u) \in Q \quad \textit{for all } u \in U \tag{2}$$

and such that

$$\nabla\hat{f} : U \longrightarrow Q \textit{ is differentiable with locally Hölder continuous derivative.}$$

6) *There are $\gamma, \delta, C_M > 0$ such that*

$$\operatorname{Prox}_{\varphi_\gamma} : Q \to U \text{ is semismooth at } \bar{q} := -\frac{1}{\gamma}\nabla\hat{f}(\bar{u})$$

and such that for each $q \in \mathbb{B}_\delta(\bar{q})$ and all $M \in \partial\operatorname{Prox}_{\varphi_\gamma}(q)$ there holds

$$\|M\|_{\mathcal{L}(Q,U)} \leq C_M.$$

7) *Defining*

$$H : Q \to Q, \qquad H(q) := \nabla\hat{f}(\operatorname{Prox}_{\varphi_\gamma}(q)) + \gamma q \tag{3}$$

with generalized derivative

$$\partial H : Q \rightrightarrows \mathcal{L}(Q,Q), \qquad \partial H(q) := \left\{\nabla^2\hat{f}(\bar{u}) \circ M + \gamma I : \; M \in \partial\operatorname{Prox}_{\varphi_\gamma}(q)\right\} \tag{4}$$

there are $\bar{\delta}, C_{\bar{M}^{-1}} > 0$ such that for each $q \in \mathbb{B}_{\bar{\delta}}(\bar{q})$ all $\bar{M} \in \partial H(q)$ are invertible and satisfy

$$\|\bar{M}^{-1}\|_{\mathcal{L}(Q,Q)} \leq C_{\bar{M}^{-1}}.$$

Remark 1 Note that $\operatorname{Prox}_{\varphi_\gamma}$ is an operator from U to U, but is required to be semismooth from Q to U in 6). Note, furthermore, that $\bar{q} \in Q$ holds in 6) due to (2).

Remark 2 Under Assumption 1 there are constants $L_P, L_\nabla > 0$ such that

$$\|\operatorname{Prox}_{\varphi_\gamma}(q) - \operatorname{Prox}_{\varphi_\gamma}(\bar{q})\|_U \leq L_P\|q - \bar{q}\|_Q \quad \text{and} \quad \|\nabla\hat{f}(u) - \nabla\hat{f}(\bar{u})\|_Q \leq L_\nabla\|u - \bar{u}\|_U$$

are satisfied for all q close to $\bar{q}$, respectively, for all u close to $\bar{u}$. The constants L_P and L_∇ will appear in the convergence results below.

Remark 3 It would be enough to require 3)–5) only locally around $\bar{u}$.

Since f can be nonconvex and since φ can be nonsmooth, (P) is a *nonconvex* and *nonsmooth* optimization problem, in general. It may also feature a convex admissible set, as φ is extended real-valued. We tackle (P) by reformulating its first order optimality condition as operator equation $H(q) = 0$. The approach to use Robinson's normal map [20] for the reformulation is inspired by [19, Section 3].

Lemma 1 *Let Assumption 1 hold. Then $\bar{u}$ satisfies the necessary optimality condition $0 \in \nabla f(\bar{u}) + \partial\varphi(\bar{u})$ of (P) and $\bar{q}$ satisfies $H(\bar{q}) = 0$, where H is given by (3). Moreover, for any $\hat{q} \in Q$ with $H(\hat{q}) = 0$ the point $\hat{u} := \operatorname{Prox}_{\varphi_\gamma}(\hat{q})$ satisfies $0 \in \nabla f(\hat{u}) + \partial\varphi(\hat{u})$. If the objective in (P) is convex, then any such $\hat{u}$ is a global solution of (P).*

Proof It is well-known that the local solution $\bar{u}$ of (P) satisfies $0 \in \nabla f(\bar{u}) + \partial\varphi(\bar{u})$. Since $\bar{q} = -\frac{1}{\gamma}\nabla\hat{f}(\bar{u})$ by definition, we obtain $\gamma(\bar{q} - \bar{u}) = -\nabla f(\bar{u}) \in \partial\varphi(\bar{u})$, hence $\bar{u} = \operatorname{Prox}_{\varphi_\gamma}(\bar{q})$ by (1). Inserting this into $\bar{q} = -\frac{1}{\gamma}\nabla\hat{f}(\bar{u})$ implies $H(\bar{q}) = 0$.

If $\hat{q} \in Q$ with $H(\hat{q}) = 0$ is given and we set $\hat{u} := \operatorname{Prox}_{\varphi_\gamma}(\hat{q}) \in U$, then we have

$$H(\hat{q}) = 0 \quad \Longrightarrow \quad -\nabla\hat{f}(\hat{u}) - \gamma\hat{u} = \gamma\left(\hat{q} - \hat{u}\right) \quad \Longrightarrow \quad -\nabla f(\hat{u}) \in \partial\varphi(\hat{u}),$$

where the final implication involves (1). The last identity yields $0 \in \nabla f(\hat{u}) + \partial\varphi(\hat{u})$.

The assertion concerning convexity is true since it just restates that the necessary optimality condition is sufficient for global optimality in convex optimization.

Next we provide the new algorithm. It aims at solving the operator equation $H(q) = 0$. Note that H acts on the artificial variable q that is related to the control u by $u^k = \mathrm{Prox}_{\varphi_\gamma}(q^k)$ for $k \geq 1$, respectively, $\bar{q} = -\frac{1}{\gamma}\nabla\hat{f}(\bar{u})$.

The key idea of the new method is to replace the Hessian $\nabla^2\hat{f}(u^k)$ that appears in semismooth Newton methods by a quasi-Newton approximation B_k. In contrast, the generalized derivative of the proximal map $\mathrm{Prox}_{\varphi_\gamma}$ is left unchanged. The algorithm thus combines a quasi-Newton method with a semismooth Newton method and can be regarded as a hybrid approach. It reads as follows.

Algorithm 1: Hybrid semismooth quasi-Newton method for (P)

```
   Input: $u^0 \in U$,  $B_0 \in \mathcal{L}(U,Q)$,  $0 \leq \sigma_{\min} \leq \sigma_{\max} \leq 2$,  $\gamma > 0$
1  Let $q^0 := -\frac{1}{\gamma}\nabla f(u^0)$
2  for $k = 0, 1, 2, \ldots$ do
3      if $H(q^k) = 0$ then let $\bar{u} := \mathrm{Prox}_{\varphi_\gamma}(q^k)$;  STOP
4      Choose $M_k \in \partial\mathrm{Prox}_{\varphi_\gamma}(q^k)$
5      Let $\tilde{M}_k := B_k M_k + \gamma I$
6      Solve $\tilde{M}_k s^k = -H(q^k)$ for $s^k$
7      Let $q^{k+1} := q^k + s^k$ and $u^{k+1} := \mathrm{Prox}_{\varphi_\gamma}(q^{k+1})$
8      Let $s_u^k := u^{k+1} - u^k$ and $y^k := \nabla\hat{f}(u^{k+1}) - \nabla\hat{f}(u^k)$
9      Choose $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$
10     if $s_u^k \neq 0$ then let $B_{k+1} := B_k + \sigma_k(y^k - B_k s_u^k)\frac{(s_u^k,\cdot)_U}{\|s_u^k\|_U^2}$;
11     else let $B_{k+1} := B_k$
12 end
   Output: $\bar{u}$
```

In the numerical study in Section 6 we work exclusively with $(\sigma_k) \equiv 1$ in line 9, i.e., the classical Broyden update, as already this simple choice results in efficient algorithms. Also, we will compare this update to other update formulas, specifically

$$B_{k+1} = B_k + \left(y^k - B_k s_u^k\right)\frac{\left(y^k - B_k s_u^k, \cdot\right)_U}{\left(y^k - B_k s_u^k, s_u^k\right)_U} \tag{SR1}$$

and

$$B_{k+1} = B_k + y^k\frac{\left(y^k, \cdot\right)_U}{\left(y^k, s_u^k\right)_U} - B_k s_u^k\frac{\left(B_k s_u^k, \cdot\right)_U}{\left(B_k s_u^k, s_u^k\right)_U}. \tag{BFGS}$$

Moreover, it should be clear that Algorithm 1 needs to be globalized for the numerical experiments. We will observe in the first half of Section 6 that if $\hat{f}$ is convex, then a simple line search suffices for this. In contrast, if (P) is severely nonlinear, then a trust-region globalization yields better results, cf. the optimal control of the Bloch equation in the second half of Section 6.

3.2 Convergence results

For the iterates (q^k) of Algorithm 1 we have the following convergence result.

Theorem 1 *Let Assumption 1 hold and let $\beta \in (0,1)$. Then:*

1) *There exists $\varepsilon > 0$ such that for every initial pair $(u^0, B_0) \in U \times \mathcal{L}(U,Q)$ with $\|u^0 - \bar{u}\|_U < \varepsilon$ and $\|B_0 - \nabla^2 \hat{f}(u^0)\|_{\mathcal{L}(U,Q)} < \varepsilon$, Algorithm 1 either terminates after finitely many iterations or generates a sequence of iterates (q^k) that converges q-linearly with rate β to $\bar{q}$. If, in addition, $\sigma_{\min}, \sigma_{\max} \in (0,2)$ in Algorithm 1 and $B_0 - \nabla^2 \hat{f}(\bar{u})$ is compact, then the convergence is q-superlinear.*
2) *If $\nabla^2 \hat{f}(u^0) - \nabla^2 \hat{f}(\bar{u})$ is compact, then the compactness of $B_0 - \nabla^2 \hat{f}(\bar{u})$ in 1) can be replaced by the compactness of $B_0 - \nabla^2 \hat{f}(u^0)$.*

Proof This follows from [15, Theorem 4.2 and Theorem 4.18] for $F(u) := \nabla \hat{f}(u)$, $G(q) := \mathrm{Prox}_{\varphi_\gamma}(q)$, $\hat{G}(q) := \gamma q$, and $V := Q$. We remark that the results in [15] require (q^0, B_0) to be close to $(\bar{q}, \nabla^2 \hat{f}(\bar{u}))$, but this is implied by the fact that (u^0, B_0) is close to $(\bar{u}, \nabla^2 \hat{f}(\bar{u}))$ since $\|q^0 - \bar{q}\|_Q = \|\frac{1}{\gamma}\nabla \hat{f}(u^0) - \frac{1}{\gamma}\nabla \hat{f}(\bar{u})\|_Q \leq \frac{L_\nabla}{\gamma}\|u^0 - \bar{u}\|_U$.

Regarding convergence of (u^k), $(\nabla \hat{f}(u^k))$ and $(H(q^k))$ we obtain the following.

Corollary 1 *Let Assumption 1 hold and let (q^k) be generated by Algorithm 1. If (q^k) converges q-linearly (q-superlinearly) to $\bar{q}$, then:*

1) *(u^k) converges r-linearly (r-superlinearly) to $\bar{u}$ and satisfies, for all k sufficiently large, $\|u^k - \bar{u}\|_U \leq L_P \|q^k - \bar{q}\|_Q$.*
2) *$(\nabla \hat{f}(u^k))$ converges r-linearly (r-superlinearly) to $\nabla \hat{f}(\bar{u})$ and satisfies, for all k sufficiently large, $\|\nabla \hat{f}(u^k) - \nabla \hat{f}(\bar{u})\|_Q \leq L_\nabla \|u^k - \bar{u}\|_U$ and $\|\nabla \hat{f}(u^k) - \nabla \hat{f}(\bar{u})\|_Q \leq L_\nabla L_P \|q^k - \bar{q}\|_Q$.*
3) *$(H(q^k))$ converges r-linearly (q-superlinearly) to zero and satisfies, for all k sufficiently large, $\|H(q^k)\|_Q \leq (L_\nabla L_P + \gamma)\|q^k - \bar{q}\|_Q$.*

Proof This follows from [15, Corollary 4.4 and Corollary 4.20] for $F(u) := \nabla \hat{f}(u)$, $G(q) := \mathrm{Prox}_{\varphi_\gamma}(q)$, $\hat{G}(q) := \gamma q$, and $V := Q$.

3.3 A general approach for local uniform invertibility

The following result is taken from [19, Section 3]. It allows to conveniently establish condition 7) of Assumption 1.

Lemma 2 *Let conditions 1)–6) of Assumption 1 hold and let H and ∂H be given by (3), respectively, (4). Moreover, suppose that $\partial \mathrm{Prox}_{\varphi_\gamma}$ can be extended to U in such a way that $\partial \mathrm{Prox}_{\varphi_\gamma}(u) \subset \mathcal{L}(U,U)$ for every $u \in U$. Then condition 7) of Assumption 1 is satisfied if there exist $\nu, \delta > 0$ such that for each $u \in \mathbb{B}_\delta(\bar{u})$ and all $M \in \partial \mathrm{Prox}_{\varphi_\gamma}(u)$ we have*

- $\|M\|_{\mathcal{L}(U,U)} \leq 1$,
- $(Mh, h)_U \geq 0$ *for all $h \in U$,*

- $(Mh_1, h_2)_U = (h_1, Mh_2)_U$ *for all* $h_1, h_2 \in U$,
- *and*

$$\gamma\,(h, Mh)_U + \left(\nabla^2 \hat{f}(\bar{u})Mh, Mh\right)_U \geq \nu\,(h, Mh)_U \qquad \text{for all } h \in U. \tag{5}$$

Proof This is [19, Lemma 3.15] for the situation at hand.

Remark 4 Inequality (5) is, in particular, valid if $\nabla^2 \hat{f}(\bar{u})$ is positive semidefinite.

4 Application to PDE-constrained optimal control

In this section we show for a model problem how the hybrid approach can be applied to PDE-constrained optimal control problems. These problems are well-suited for the application of the hybrid method because the assumptions for fast local convergence are typically satisfied.

4.1 An important proximal map

To facilitate the discussion of the model problem in Section 4.2, we study the associated proximal map in this section. To this end, let $N \in \mathbb{N}$, $T > 0$, and $U := L^2(I)^N$, where $I := (0, T)$ for some $T > 0$. We are interested in the proximal map of

$$\varphi : U \longrightarrow \mathbb{R} \cup \{+\infty\}, \qquad \varphi(u) := \delta_{U_{\text{ad}}}(u) + \sum_{i=1}^{N} \beta_i \|u_i\|_{L^1(I)},$$

where β_i, $1 \leq i \leq N$, are nonnegative real numbers, U_{ad} is given by

$$U_{\text{ad}} := \left\{u = (u_1, \dots, u_N)^T \in U : \; a_i \leq u_i \leq b_i \text{ a.e. in } I, \; 1 \leq i \leq N\right\}$$

for functions $a, b \in L^\infty(I)^N$ that satisfy $a \leq b$ a.e. in I (the inequality is meant componentwise), and $\delta_{U_{\text{ad}}} : U \longrightarrow \{0, +\infty\}$ denotes the characteristic function of U_{ad}. For reasons that become clear in Section 4.2, we fix positive weights $\alpha_1, \dots, \alpha_N$ and endow U with the norm $\|u\|_U := (\sum_{i=1}^N \alpha_i \|u_i\|_{L^2(I)}^2)^{1/2}$. This norm is equivalent to the standard norm in U and it is derived from a scalar product, hence U is a Hilbert space with respect to it. We write $\Pi_{U_{\text{ad}}} : U \longrightarrow U$ for the projection onto U_{ad}, i.e.,

$$\Pi_{U_{\text{ad}}}(u) := \max(a, \min(u, b)) \quad \text{for a.e. } x \in \Omega.$$

Furthermore, let us introduce the *soft-shrinkage operator* $\sigma : U \longrightarrow U$, which is given componentwise for $1 \leq i \leq N$ and constants $\alpha_i > 0$ and $\beta_i \geq 0$ by

$$\sigma_i(u)(t) := \rho_i(u_i(t)) \qquad \text{for} \qquad \rho_i : \mathbb{R} \longrightarrow \mathbb{R}, \quad \rho_i(s) := \left(s - \frac{\beta_i}{\alpha_i}\right)^+ + \left(s + \frac{\beta_i}{\alpha_i}\right)^-,$$

where $(r)^+ := \max(0, r)$ and $(r)^- := \min(0, r)$ for all $r \in \mathbb{R}$.

The proximal map for $\gamma = 1$ can now be characterized as follows.

Lemma 3 *The proximal map* $\operatorname{Prox}_{\varphi_1} : U \longrightarrow U_{\text{ad}}$ *is given by* $\operatorname{Prox}_{\varphi_1} = \Pi_{U_{\text{ad}}} \circ \sigma$.

Proof This can be established as in [19, Section 3.3] or through direct computation.

It is important to note that $\operatorname{Prox}_{\varphi_1}$ is semismooth.

Lemma 4 $\operatorname{Prox}_{\varphi_1}$ *is semismooth at every $q \in Q$ when considered as a mapping from $Q := C([0,T])^N$ to U with respect to the generalized derivative $\partial \operatorname{Prox}_{\varphi_1}(q) \subset \mathcal{L}(Q,U)$ given by*

$$\partial \operatorname{Prox}_{\varphi_1}(q) := \bigcup_{\substack{r \in L^\infty(I)^N \text{ with} \\ 0 \le r \le 1 \text{ a.e. in } I}} \left\{ M(q,r) \right\},$$

where $0 \le r \le 1$ is meant componentwise and $M = M(q,r) \in \mathcal{L}(Q,U)$ is for $(q,r) \in Q \times L^\infty(I)^N$ defined as

$$(Mh)_i(t) := \begin{cases} h_i(t) & \text{if } |q_i(t)| > \frac{\beta_i}{\alpha_i} \ \wedge \ \sigma_i(q)(t) \in (a_i(t), b_i(t)), \\ 0 & \text{if } |q_i(t)| < \frac{\beta_i}{\alpha_i} \ \vee \ \sigma_i(q)(t) \notin [a_i(t), b_i(t)], \\ r_i(t)h_i(t) & \text{else} \end{cases} \tag{6}$$

if $i \in \{1, \ldots, N\}$ is such that $\beta_i > 0$. If $\beta_i = 0$, then the conditions involving $\frac{\beta_i}{\alpha_i}$ have to be dropped in (6) and there holds $\sigma_i(q) = q_i$. In any case, $\|M\|_{\mathcal{L}(Q,U)} \le T^{\frac{1}{2}} (\sum_{i=1}^N \alpha_i)^{\frac{1}{2}}$ holds for each $q \in Q$ and all $M \in \partial \operatorname{Prox}_{\varphi_1}(q)$.

Proof Since $\operatorname{Prox}_{\varphi_1} = \Pi_{U_{\mathrm{ad}}} \circ \sigma$, respectively, $\operatorname{Prox}_{\varphi_1} = \Pi_{U_{\mathrm{ad}}}$ is a superposition operator, the representation (8) can, for instance, be deduced from [12, Theorem 2.13]. The estimate for $\|M\|_{\mathcal{L}(Q,U)}$ follows since for $h \in Q$ with $\|h\|_Q \le 1$ we have $(Mh)_i(t) \le 1$ for a.e. $t \in I$ and all $1 \le i \le N$.

Remark 5 We emphasize that the superposition operator $\Pi_{U_{\mathrm{ad}}} \circ \sigma$ is not semismooth from U to U, cf. [24, 27]. Correspondingly, the role of the additional space Q in Assumption 1 is to ensure the necessary norm gap. In turn, the demand for $\nabla \hat{f}$ to map U to Q, cf. 5) of Assumption 1, is a *smoothing property.*

4.2 Time-dependent control of the heat equation

Let $N \in \mathbb{N}$, $Q := C([0,T])^N$, $U := L^2(I)^N$, and $Y := P := W(I; L^2(\Omega), H_0^1(\Omega))$. We consider the optimal tracking of the linear heat equation in $I \times \Omega$ with N time-dependent controls $u(t) = (u_1(t), \ldots, u_N(t))^T$, where $\Omega \subset \mathbb{R}^d$, $1 \le d \le 3$, is a nonempty and bounded Lipschitz domain, and the time domain is $I := (0,T)$ for a fixed final time $T > 0$:

$$\min_{(y,u) \in Y \times U_{\mathrm{ad}}} \frac{1}{2}\|y - y_{\mathrm{d}}\|^2_{L^2(I \times \Omega_{\mathrm{obs}})} + \sum_{i=1}^N \frac{\alpha_i}{2} \|u_i\|^2_{L^2(I)} + \sum_{i=1}^N \beta_i \|u_i\|_{L^1(I)}$$

$$\text{s. t.} \quad \begin{cases} y_t - \Delta y = \sum_{i=1}^N g_i(x) u_i(t) & \text{in } I \times \Omega, \\ y = 0 & \text{on } \Sigma, \\ y(0,x) = y_0(x) & \text{in } \Omega, \end{cases} \tag{OCP}$$

where $\Sigma := I \times \partial\Omega$. Moreover, $y_\mathrm{d} \in L^2(I \times \Omega_\mathrm{obs})$ is the desired state, $\Omega_\mathrm{obs} \subset \Omega$ is the observation domain, $\alpha_i > 0$ are the control cost parameters per control function, $\beta_i \geq 0$ influences the size of the support of u_i, $y_0 \in L^2(\Omega)$ is the initial state, and $g_i \in L^2(\Omega)$ are fixed spatial functions whose support is denoted by $\omega_i \subset \Omega$, $1 \leq i \leq N$. For instance, g_i could be the characteristic function χ_{ω_i} of a given control domain $\omega_i \subset \Omega$. The set of admissible controls is given by

$$U_\mathrm{ad} := \left\{u = (u_1, \ldots, u_N)^T \in U : \; a_i \leq u_i \leq b_i \text{ a.e. in } I,\; 1 \leq i \leq N\right\} \tag{7}$$

with functions $a, b \in L^\infty(I)^N$ that satisfy $a \leq b$ a.e. in I. By using the same norm on U as in Section 4.1 we can regard (OCP) as a special instance of (P) with $\gamma = 1$. From [26, Theorem 3.13] we obtain that for every $u \in U$ there exists a unique $y = y(u) \in Y$ such that the PDE-constraints in (OCP) are satisfied; the dependence $u \mapsto y(u)$ is linear and continuous from U to Y. This implies that the solution operator $u \mapsto y(u)$ is infinitely many times continuously differentiable. Thus, $\hat{f}(u) := \frac{1}{2}\|y(u) - y_\mathrm{d}\|^2_{L^2(I\times\Omega_\mathrm{obs})}$ is continuously differentiable from U to $\mathbb{R}$. Since the control reduced version (P) of (OCP) is a convex problem with strongly convex objective, it is standard to show that it possesses a unique solution $\bar{u} \in U_\mathrm{ad}$; the associated state is denoted by $\bar{y} := y(\bar{u}) \in Y$. We can now derive the following result.

Lemma 5 *For the control reduced version of* (OCP) *the mapping H defined in* (3) *is for $\gamma = 1$ given by*

$$H : Q \to Q, \qquad H_i(q)(t) = \int_{\omega_i} g_i(x) p\Big(\Pi_{U_\mathrm{ad}}(\sigma(q))\Big)(t,x)\,dx + \alpha_i q_i(t), \qquad 1 \leq i \leq N. \tag{8}$$

Here, $p = p(u) \in P$ is the adjoint state*, i.e., the unique solution of the* adjoint equation

$$\begin{cases} -p_t - \Delta p = \chi_{I\times\Omega_\mathrm{obs}} \cdot (y(u) - y_d) & \text{in } I \times \Omega, \\ \qquad\quad p = 0 & \text{on } \Sigma, \\ \qquad p(T) = 0 & \text{in } \Omega. \end{cases} \tag{9}$$

Moreover, H has a unique root $\bar{q} \in Q$.

Proof Adjoint calculus yields $(\nabla\hat{f}(u))_i(t) = \int_{\omega_i} g_i(x)p(t,x)/\alpha_i \,\mathrm{d}x$ for $1 \leq i \leq N$, where $p = p(u)$ is the adjoint state. Inserting this in $H(q) = \nabla\hat{f}(\mathrm{Prox}_{\varphi_1}(q)) + q$ yields (8) due to Lemma 3. Moreover, the assumptions on the problem data imply $p(u) \in P$ for the solution of the adjoint equation, cf. [26, Lemma 3.17]. To show that H maps to Q we deduce from the continuous embedding $P \hookrightarrow C([0,T]; L^2(\Omega))$, cf. [7, Theorem 11.4], that $p(u) \in C([0,T]; L^2(\Omega))$. Therefore, defining $\lambda = \lambda(u)$ by $\lambda_i(t) := \int_{\omega_i} g_i(x)p(t,x)/\alpha_i \,\mathrm{d}x$, $t \in [0,T]$, where $1 \leq i \leq N$, implies $\lambda \in Q$. Since $\lambda(u) = \nabla\hat{f}(u)$, it follows that $\nabla\hat{f}$ maps U to Q and hence that H maps to Q. From Lemma 1 we infer by convexity that the unique solution $\bar{u}$ of the control reduced version of (OCP) corresponds to a unique root $\bar{q}$ of H.

Remark 6 The proof of Lemma 5 demonstrates that we have to choose Q in such a way that $t \mapsto \int_{\omega_i} g_i(x)p(t,x)\,\mathrm{d}x$ belongs to Q, where p solves (9). Thus, the available regularity of the adjoint state p, respectively, of the multiplier λ, restricts the choice of Q. If additional regularity is available, then Q may be chosen as a space of smoother functions than $C([0,T])^N$. For instance, from [12, Theorem 1.39] we deduce that if $\Omega_{\text{obs}} = \Omega$ and $y_d \in Y$, then there holds $\frac{\partial p(u)}{\partial t} \in W(I; L^2(\Omega), H_0^1(\Omega))$, hence $p(u) \in H^1(I; H_0^1(\Omega))$. This implies $\lambda \in Q$ for the choice $Q := H^1(I)^N$. In fact, using $W(I; L^2(\Omega), H_0^1(\Omega)) \hookrightarrow C([0,T]; L^2(\Omega))$ we obtain $p \in C^1([0,T]; L^2(\Omega))$, which implies $\lambda \in Q$ for $Q := C^1([0,T])^N$. We stress that Lemma 5 is valid for all these choices of Q.

Assumption 1 holds unconditionally for (OCP).

Lemma 6 *The control reduced version of* (OCP) *fulfills Assumption 1 with* $H : Q \to Q$ *given by* (8).

Proof Conditions 1)–4) of Assumption 1 were already established, cf. the remarks above Lemma 5. In the proof of Lemma 5 we have demonstrated that $\nabla \hat{f}$ maps U to Q. Since $\nabla \hat{f} : U \to Q$ is linear and continuous, it is infinitely many times continuously differentiable. This yields 5). Condition 6) follows from Lemma 4. To establish 7) we use Lemma 2. The representation in Lemma 4 shows that $\partial \operatorname{Prox}_{\varphi_1}(q)$, $q \in Q$, can be extended in a canonical way to $\partial \operatorname{Prox}_{\varphi_1}(u)$, $u \in U$. From the linearity of $u \mapsto y(u)$ we obtain that $\hat{f}(u) = \frac{1}{2}\|y(u) - y_d\|^2_{L^2(I \times \Omega_{\text{obs}})}$ is convex, hence (5) is fulfilled. Also, we readily check that the first three properties listed in Lemma 2 are satisfied by the elements of $\partial \operatorname{Prox}_{\varphi_1}(u)$, $u \in U$. Thus, 7) holds.

Remark 7 If $\Omega_{\text{obs}} = \Omega$ and $y_d \in Y$, then Lemma 6 is also true for the choices $Q := H^1(I)^N$ and $P := H^1(I; H_0^1(\Omega))$ as well as $Q := C^1([0,T])^N$ and $P := C^1([0,T]; L^2(\Omega))$, cf. Remark 6.

We obtain the following convergence result for Algorithm 1, in which we write $\nabla^2 \hat{f}$ for the constant Hessian. Note in 2) and 3) that (u^k) converges in various norms.

Theorem 2 *1) Let $(\bar{y}, \bar{u}) \in Y \times U_{\text{ad}}$ be the solution of* (OCP)*, let H be given by* (8)*, and denote by $\bar{q} \in Q$ the unique root of H. Moreover, let $\beta \in (0,1)$. Then there exists $\varepsilon > 0$ such that for every initial pair $(u^0, B_0) \in U \times \mathcal{L}(U,Q)$ with $\|u^0 - \bar{u}\|_U < \epsilon$ and $\|B_0 - \nabla^2 \hat{f}\|_{\mathcal{L}(U,Q)} < \varepsilon$, Algorithm 1 either terminates after finitely many iterations or generates a sequence of iterates (q^k) that converges q-linearly with rate β to $\bar{q}$ in Q. If, in addition, $\sigma_{\min}, \sigma_{\max} \in (0,2)$ in Algorithm 1 and $(B_0 - \nabla^2 \hat{f}) \in \mathcal{L}(U,Q)$ is compact, then the convergence is q-superlinear.*

2) If (q^k) is generated by Algorithm 1, then $(u^k)_{k \geq 1} \subset U_{\text{ad}}$, i.e., every u^k except possibly the starting point u^0 is feasible for (OCP)*. Moreover, $(u^k)_{k\geq 1}, \{\bar{u}\} \subset L^\infty(I)^N$ and there are $L_y, L_p > 0$ such that*

$$\|u^k - \bar{u}\|_{L^s(I)^N} \leq \|q^k - \bar{q}\|_{L^s(I)^N}, \qquad \|u^k - \bar{u}\|_{L^2(I)^N} \leq T^{\frac{1}{2}} \|q^k - \bar{q}\|_{C([0,T])^N},$$
$$\|y(u^k) - \bar{y}\|_Y \leq L_y \|q^k - \bar{q}\|_Q, \qquad \|p(u^k) - \bar{p}\|_P \leq L_p \|q^k - \bar{q}\|_Q$$

hold for all $k \geq 1$ and all $s \in [1,\infty]$.

If, in addition, $a, b \in Q$ holds, then we have $(u^k)_{k\geq 1}, \{\bar{u}\} \subset Q$ and for all $k \geq 1$

$$\|u^k - \bar{u}\|_Q \leq \|q^k - \bar{q}\|_Q. \tag{10}$$

In particular, $\|q^k - \bar{q}\|_Q \to 0$ for $k \to \infty$ implies $\|u^k - \bar{u}\|_Q \to 0$.

3) If (q^k) is generated by Algorithm 1 and converges q-linearly (q-superlinearly) to $\bar{q}$ in Q, then (u^k), $(y(u^k))$ and $(p(u^k))$ converge r-linearly (r-superlinearly) in $L^s(I)^N$, respectively, Y and P, where s is as in 2). Moreover, $(H(q^k))$ converges r-linearly (q-superlinearly) in Q to zero, then.

Proof **Proof of 1):** The claim of 1) follows from Theorem 1, part 1), which can be applied since Assumption 1 is satisfied, cf. Lemma 6.

Proof of 2): The feasibility of the u^k is valid since $\mathrm{Prox}_{\varphi_1}(Q) \subset U_{\mathrm{ad}}$ and since $u^k = \mathrm{Prox}_{\varphi_1}(q^k)$ for $k \geq 1$. Moreover, the property $(u^k)_{k\geq 1}, \{\bar{u}\} \subset L^\infty(I)^N$ follows from $U_{\mathrm{ad}} \subset L^\infty(I)^N$. To show the first inequality, we remark that $q^k, \bar{q} \in Q$ implies $q^k, \bar{q} \in L^s(I)^N$ for all $s \in [1, \infty]$ and all $k \geq 0$. It is straightforward to infer for $q, \bar{q} \in L^s(I)^N$ that $|\rho_i(q_i(t)) - \rho_i(\bar{q}_i(t))| \leq |q_i(t) - \bar{q}_i(t)|$ for a.e. $t \in I$, $1 \leq i \leq N$. The same can be established for ρ_i replaced by $(\Pi_{U_{\mathrm{ad}}})_i$. Together, this implies $|u_i(t) - \bar{u}_i(t)| \leq |q_i(t) - \bar{q}_i(t)|$ for a.e. $t \in I$, $1 \leq i \leq N$, proving the first error bound in 2). Moreover, this also implies (10) provided that $(\Pi_{U_{\mathrm{ad}}} \circ \sigma)(Q) \subset Q$ if $a, b \in Q$. This property of $\Pi_{U_{\mathrm{ad}}} \circ \sigma$ is elementary to see, for instance by showing it separately for σ and $\Pi_{U_{\mathrm{ad}}}$. The second error bound follows from the first for $s = 2$ by use of $\|q^k - \bar{q}\|_{L^2(I)^N} \leq |I|^{\frac{1}{2}} \|q^k - \bar{q}\|_{C([0,T])^N}$, where $|I|$ is the Lebesgue measure of I. Since $u \mapsto y(u)$ and $u \mapsto p(u)$ are linear and continuous from U to $Y = P$, they are globally Lipschitz, too. This in combination with the estimate $\|u^k - \bar{u}\|_{L^2(I)^N} \leq \|q^k - \bar{q}\|_{L^2(I)^N}$ and the continuous embedding $Q \hookrightarrow L^2(I)^N$ yields the third and fourth error bound.

Proof of 3): The claims follow from the error estimates in 2) and, for $(H(q^k))$, from part 3) of Corollary 1.

Remark 8 It is not difficult to argue that Theorem 2 also holds for $Q := L^{\hat{s}}(I)^N$ for any $\hat{s} \in (2, \infty]$. Of course, the L^s estimates of that theorem are then only true for $s \in [1, \hat{s}]$. The use of a weaker norm in Q relaxes the assumption on (u^0, B_0) and the compactness requirement in that theorem.

In Theorem 2 we have worked with $Q = C([0,T])^N$ and $P = W(I; L^2(\Omega), H_0^1(\Omega))$. If $\Omega_{\mathrm{obs}} = \Omega$ and y_d is more regular, then we can employ stronger spaces, resulting in stronger convergence properties, but also in the need for better initial data (u^0, B_0). For simplicity we consider constant bounds.

Lemma 7 *Let $\Omega_{\mathrm{obs}} = \Omega$, $y_d \in Y$, and let a_i, b_i be constant for each $1 \leq i \leq N$. Then all claims of Theorem 2 except (10) are true for $Q := H^1(I)^N$, $U := L^2(I)^N$, $Y := W(I; L^2(\Omega), H_0^1(\Omega))$, and $P := H^1(I; H_0^1(\Omega))$. If $Q := L^\infty(I)$ is used instead, then all claims of Theorem 2 except for the second error estimate in part 2) are true and the operator $\nabla^2 \hat{f} \in \mathcal{L}(U, Q)$ is compact.*

Proof **Proof for $\mathbf{Q := H^1(I)^N}$:** The proof is completely analogue to the one of Theorem 2, except for the claim above (10) and the one below (10). That is, we have to establish that $(u^k)_{k\geq 1}, \{\bar{u}\} \subset Q = H^1(I)^N$ and $u^k \to \bar{u}$ in Q provided that $q^k \to \bar{q}$

in Q. In fact, this follows since $\mathrm{Prox}_{\varphi_1} = \Pi_{U_{\mathrm{ad}}} \circ \sigma$ satisfies $\mathrm{Prox}_{\varphi_1}(Q) \subset Q$ and since $\mathrm{Prox}_{\varphi_1} : Q \longrightarrow Q$ is continuous. Both properties can be proven separately for $\Pi_{U_{\mathrm{ad}}}$ and σ. For $\Pi_{U_{\mathrm{ad}}}$ these properties follow from [14, Chapter II, Corollary A.5], which shows that $\Pi_{U_{\mathrm{ad}}}(W^{1,s}(\Omega)) \subset W^{1,s}(\Omega)$ for all $s \in [1, \infty]$, and from [2, Theorem 9.5]. Using the fact that cut-off does not increase the H^1-norm, cf. [14, Theorem A.1], the corresponding proof for σ is elementary.
Proof for $\mathbf{Q := L^\infty(I)^N}$: The compactness of $\nabla^2 \hat{f}$ follows from the compactness of $H^1(I)^N \hookrightarrow L^\infty(I)^N$. All other claims can be established as in the proof of Theorem 2.

Still higher regularity is available in the situation of Lemma 7.

Lemma 8 *Let $\Omega_{\mathrm{obs}} = \Omega$, $y_d \in Y$, and let a_i, b_i be constant for each $1 \leq i \leq N$. Then all claims of Theorem 2 except (10) and the convergence assertion below it are true for $Q := C^{0,\hat{s}}([0,T])^N$ with arbitrary $\hat{s} \in (0,1]$, $U := L^2(I)^N$, $Y := W(I; L^2(\Omega), H_0^1(\Omega))$, and $P := C^1([0,T]; L^2(\Omega))$. In this setting $q^k \longrightarrow \bar{q}$ in Q implies $u^k \longrightarrow \bar{u}$ in $C^{0,\hat{s}-\tau}([0,T])^N$ for any $\tau \in (0, \hat{s})$. Moreover, if $\hat{s} \neq 1$, then $\nabla^2 \hat{f} \in \mathcal{L}(U, Q)$ is compact.*

Proof The main task is to establish that $\mathrm{Prox}_{\varphi_1} = \Pi_{U_{\mathrm{ad}}} \circ \sigma$ satisfies $\mathrm{Prox}_{\varphi_1}(Q) \subset Q$ and that $\mathrm{Prox}_{\varphi_1} : Q \longrightarrow C^{0,\hat{s}-\tau}([0,T])^N$ is continuous at $\bar{q}$ for any $\tau \in (0, \hat{s})$. The proof can be undertaken separately for $\Pi_{U_{\mathrm{ad}}}$ and σ. The property $\mathrm{Prox}_{\varphi_1}(Q) \subset Q$ follows from [2, Theorem 7.1], and the continuity at $\bar{q}$ is a consequence of [2, first part of Theorem 7.6]. The compactness of $\nabla^2 \hat{f} \in \mathcal{L}(U, Q)$ is implied by the fact that $C^{0,1}([0,T])^N \hookrightarrow C^{0,\hat{s}}([0,T])^N$ is compact for $\hat{s} \in (0,1)$.

5 Implementation

The hybrid framework is tested with three different quasi-Newton updates: Broyden, SR1, and BFGS. The methods are implemented with limited-memory techniques storing the last up to L (called the limit) updates as vectors and matrix-free. Consequently, the Newton systems are solved with iterative methods, specifically with GMRES or cg. The limited-memory BFGS method is implemented in the compact variant according to [6], see also [17, (7.24)]. Additionally, the quasi-Newton methods are compared to Newton's method itself, i.e., setting $B_k = \nabla^2 \hat{f}(u^k)$ for all k and dropping lines 8–11 in Algorithm 1. Here, the matrix-free evaluation in a direction is implemented via forward-backward solve. We stress that when Newton's method is used, Algorithm 1 is a standard semismooth Newton method.
The methods are applied with three different globalization techniques. First, a standard backtracking line search on the residual norm $\|H\|_U$ is used together with GMRES (ls-GMRES). The line search selects the smallest integer $0 \leq j \leq 18$ with $\|H(q^k + 0.5^j s^k)\|_U < \|H(q^k)\|_U$, and $j = 19$ otherwise. GMRES from Matlab is used with a tolerance of 10^{-10} and a maximum number of 50 iterations. Second, a non-monotone line search (nls-GMRES) with $N_{\mathrm{ls}} \in \mathbb{N}$ steps is used, where the step size $\rho^j = 0.5^j$, $0 \leq j \leq 18$, is accepted as soon as $R(q^k + \rho^j s^k) < \max\{R(q^k), R(q^{k-1}), \ldots, R(q^{k-N_{\mathrm{ls}}+1})\}$ holds, and $j = 19$ otherwise. Therein, R will be the residual norm $\|H(\cdot)\|_U$ or the objective $f(\cdot) + \varphi(\cdot)$ of (P). Third, a trust-region method is investigated based on Steihaug-cg (tr-cg), cf. [25]. The precise algorithm is included in Appendix A. It is

started with a radius $\varrho_0 = 0.1$ and stopped with a relative tolerance of 10^{-5}. The parameters are $\sigma_1 = 0.05, \sigma_2 = 0.25, \sigma_3 = 0.7$, radius factors $f_1 = 0.4, f_2 = 2, f_3 = 0.6$, and a maximum radius $\varrho_{\max} = 2$. Up to 300 iterations are allowed. The update of the quasi-Newton matrix is also carried out at rejected steps. Additionally, in case of BFGS, the update is only applied if the curvature condition $(y^k, s_u^k)_U > 0$ holds. To be able to use Steihaug-cg, the linear system is first reduced to a symmetric one by restricting to the Hilbert space induced by the inner product $(\cdot, M_k \cdot)_U$ with $M_k \in \partial \operatorname{Prox}_{\varphi_\gamma}(q^k)$, cf. Lemma 2 and [19, Def. 3.4]. Then a correction step gives the full update, cf. [19, (3.25)]. The cg method is limited to 100 iterations and stopped with a relative tolerance of 10^{-10}. We use small tolerances to suppress the influence of inexact linear system solves.

6 Numerical experiments

The numerical experiments below show the application of the hybrid method to optimal control problems. The first example is the heat equation, the second the bilinear control of the Bloch equation in magnetic resonance imaging. As nonsmooth problem parts we deal with pointwise box constraints on the controls and sparsity promoting objectives. All computations are carried out using Matlab 2017a and a workstation with two Intel Xeon X5675 (24GB RAM, twelve cores with 3.06GHz). All time measurements are performed on a single CPU without multi-threading.

6.1 Time-dependent tracking of the heat equation

The first example problem is sparse control of the linear heat equation with $\Omega_{\text{obs}} = \Omega = (-1, 1)^2$ and time domain $I = (0, 1)$. Specifically, we consider

$$\min_{(y,u) \in Y \times U_{\text{ad}}} \frac{1}{2} \|y - y_{\text{d}}\|^2_{L^2(I \times \Omega_{\text{obs}})} + \frac{\alpha}{2} \|u\|^2_{L^2(I)} + \beta \|u\|_{L^1(I)}$$

$$\text{s.t.} \quad \begin{cases} y_t - \Delta y = \chi_\omega(x) u(t) & \text{in } I \times \Omega, \\ \partial_\nu y = 0 & \text{on } \Sigma, \\ y(0, x) = y_0(x) & \text{in } \Omega. \end{cases}$$

Therein, $U_{\text{ad}} = \{u \in U : a \leq u(t) \leq b \text{ for a.e. } t \in I\}$ with $a = -1$ and $b = 1$, $y_{\text{d}}(t, x) = 2\sin(2\pi t)$, $\alpha > 0$, $\beta \geq 0$, $\chi_\omega(x) \in L^\infty(\Omega)$ is the characteristic function of the right half $\omega = (0, 1) \times (-1, 1)$, and $y_0 \equiv 0$. The example is a special case of (OCP) with $N = 1$, however using Neumann instead of Dirichlet boundary conditions. We emphasize that the results of Section 4.2 can also be developed for these boundary conditions. In view of Lemma 7 and Lemma 8 we expect convergence in rather strong norms. Specifically, we are interested in q-superlinear convergence of (q^k) in $H^1(I)$ and in $C^{0,s}([0, T])$ for all $s \in (0, 1)$, convergence of (u^k) in $H^1(I)$ and in $C^{0,s-\tau}([0, T])$ for all $\tau \in (0, s)$, and r-superlinear convergence of (u^k) in $L^2(I)$ and $L^\infty(I)$. Furthermore, we should be able to observe the error estimates $\|u^k - \bar{u}\|_{L^s(I)} \leq \|q^k - \bar{q}\|_{L^s(I)}$ for $s \in \{2, \infty\}$, cf. part 2) of Theorem 2. We will investigate these properties numerically.

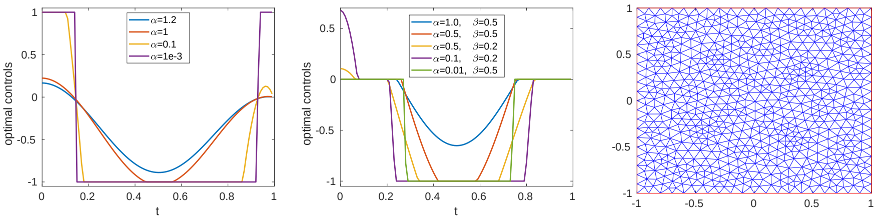

Fig. 1: Optimal control $\bar{u}$ for different α with $\beta = 0$ (left), with $\beta > 0$ (mid), and domain Ω with triangulation (right).

We use an unstructured triangular mesh with 725 P1 elements generated by Matlab's `initmesh` with `Hmax`=0.1, see Figure 1. As time-stepping scheme the CG(1) method is chosen (corresponding to the Crank-Nicolson method) with 101 equidistant time points and a piecewise constant discretization of q and u. We initialize the algorithm with $u^0 = 0$, corresponding to $q^0 = 0$. Since the problem is strongly convex, the solution is possible based on the simple globalization ls-GMRES. The optimal controls for different α, β are depicted in Figure 1. We will use varying parameters α, β, with $\alpha = 0.01$ as default value. For the limit we use $L = 30$.

The proximal mapping is $\Pi_{U_{\text{ad}}}(\sigma(q))$ from Lemma 3. In the algorithm we select the differential $M_k \in \partial(\Pi_{U_{\text{ad}}} \circ \sigma)(q^k)$ that satisfies, for all $h \in Q$, $(M_k h)(t) = h(t)$ if $|q^k(t)| \geq \beta/\alpha$ and $a < \sigma(q^k)(t) < b$, and $(M_k h)(t) = 0$ otherwise; cf. (6).

		SN	Broyden				SR1				BFGS			
α	$\lvert\mathcal{A}\rvert$		αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$	0	αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$	0	αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$	I
10^0	12	2	12	10	10	6	8	9	10	5	12	7	9	12
10^{-1}	80	4	12	15	–	7	10	17	–	6	14	11	14	18
10^{-2}	95	6	11	19	–	7	9	–	–	7	10	13	20	22
10^{-3}	99	11	10	17	–	9	9	–	–	12	9	12	12	14

Table 1: Iterations of the quasi-Newton implementations for different α (rows) and for different B_0 (columns per method) with $\beta = 0$. $|\mathcal{A}|$ is the number of control components on the bounds out of 101, $-$ means no convergence within 50 iterations.

The first study shows the performance of the optimization methods for different initializations of B_0. Table 1 shows the results for $\beta = 0$. The first two columns show α and the resulting number of active points $|\mathcal{A}|$ (optimal control on upper or lower bound) out of 101. The other columns depict the iteration numbers that are needed to reach the relative tolerance 10^{-8}, separately for semismooth Newton (SN) and the three hybrid methods. Here, four different initializations are applied, including the scaled identity αI, the zero matrix 0, and the two formulas [6, Eq.(3.23)] $B_0 = y^T s/(s^T s)I$ and [17, Eq.(7.20)] $B_0 = y^T y/(y^T s)I$. We note that both formulas are implemented in the first step with $B_0 = 0$ for Broyden/SR1 and $B_0 = \alpha I$ for BFGS. A good performance for any α and for all three methods can be obtained

by choosing $B_0 = \alpha I$. However, Broyden and SR1 show faster convergence with the zero initialization, and BFGS shows in the mean less iterations with $B_0 = y^T s/(s^T s)I$, which are the default initializations for all other studies below. We point out that from an infinite-dimensional point of view the choice $B_0 = 0$ results in $B_0 - \nabla^2 \hat{f}(\bar{u}) \in \mathcal{L}(U, Q)$ being compact, cf. Lemma 7 and Lemma 8, whereas this is not the case for the scaled identities. We recall that the compactness of $B_0 - \nabla^2 \hat{f}(\bar{u})$ is required for superlinear convergence, cf. Theorem 2, part 1).
The results for $\beta > 0$ are depicted in Table 2. Here, $|\mathcal{O}|$ collects the number of time points with zero control out of 101. We note that the inequality constraints are in general inactive ($|\mathcal{A}| = 0$) in the optimum if $\alpha \geq 1$. On the other hand, smaller values $\alpha \leq 0.01$ result in only one or two inactive points (the number of inactive points is $101 - |\mathcal{A}| - |\mathcal{O}|$). Increasing the parameter β between 0.1 and 1 increases the sparsity from around 20% to 80%. For $\beta \geq 2$ the optimal solution is zero. The four last columns show the iteration counts of the four methods with default initializations. All methods convergence quickly for all values of α and β. They tend to require fewer iterations for larger β, which corresponds to more degrees of freedom being fixed to zero. The desired tolerance is reached after at most 8 iterations for Broyden (Br), respectively, 7 iterations for SR1. BFGS needs up to 17 iterations.

α	β	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{O}\rvert$	SN	Br	SR1	BFGS
1.00	0.1	0	18	4	6	5	8
0.10	0.1	68	20	4	8	6	12
0.01	0.1	79	20	6	6	6	13
1.00	0.2	0	34	3	6	5	7
0.10	0.2	56	30	4	8	7	12
0.01	0.2	68	31	5	7	6	17
1.00	0.5	0	48	3	5	5	7
0.10	0.5	41	53	3	6	5	8
0.01	0.5	45	53	5	7	6	9
1.00	1.0	0	71	3	4	4	6
0.10	1.0	14	77	4	6	5	10
0.01	1.0	21	77	5	7	6	9

Table 2: Iteration counts for different α and β. $|\mathcal{A}|$ and $|\mathcal{O}|$ denote the number out of 101 control points that are on the bounds, respectively, that are zero.

The iteration counts of the hybrid methods for different discretizations are depicted in the upper part of Table 3 for $\alpha = 0.01, \beta = 0$ using $N_c + 1$ equidistant time points and three different meshes from `initmesh` with N_x nodes. The results show mesh independence for all three quasi-Newton methods with respect to both the spatial and the temporal discretization. The lower part of the table shows the corresponding runtimes in seconds. All values are averages of five runs. Broyden and SR1 show nearly identical runtimes in this example and are twice as fast as BFGS. All three methods outperform the semismooth Newton method in runtime. For all time and space discretizations a speedup factor of five to six is observed for Broyden and SR1, and three to four for BFGS.

iterations	SN			Broyden			SR1			BFGS		
$N_c \setminus N_x$	725	1938	7701	725	1938	7701	725	1938	7701	725	1938	7701
100	6	6	6	7	7	7	7	7	7	13	13	13
400	5	5	5	7	8	7	7	7	7	14	14	14
1600	5	5	5	7	7	7	7	7	7	14	14	14
6400	5	5	5	7	8	7	7	7	7	14	14	14
runtimes												
100	11	40	210	2	5	33	2	6	31	3	9	56
400	40	128	681	6	22	112	6	20	111	12	37	211
1600	158	486	2755	27	83	446	27	82	449	51	154	841
6400	643	1971	10855	137	456	2095	148	419	2102	290	763	3945

Table 3: Iterations and runtimes (in sec.) for different discretizations with $\alpha = 0.01, \beta = 0$. The rows show results for N_c control points. The three columns per method show results for triangulations with $N_x = 725, 1938, 7701$ nodes. All runtimes are mean values of five runs.

The next study analyzes the superlinear convergence properties numerically. For comparison the optimal solution $\bar{q}$ is first computed in high precision with the semismooth Newton method and GMRES using fine relative tolerances of 10^{-14} for both. Then the indicators of superlinear convergence

$$r_u^k := \frac{\|u^{k+1} - \bar{u}\|_Z}{\|u^k - \bar{u}\|_Z} \qquad \text{and} \qquad r_q^k := \frac{\|q^{k+1} - \bar{q}\|_Z}{\|q^k - \bar{q}\|_Z}$$

are computed for each method for the norms $Z = L^2(I), L^\infty(I)$ and the seminorms $Z = H^1(I), C^{0,1}([0,T])$. For q-superlinear convergence these indicators should converge to zero in the last steps of an optimization run. Table 4 depicts the indicators for the last four iterations. The results are obtained for $\alpha = 0.1, \beta = 0.2$, and a relative tolerance of 10^{-8}. We observe that the semismooth Newton method converges in one step as soon as the active set has converged. Broyden and SR1 show fast superlinear convergence with final indicators between $7 \cdot 10^{-3}$ and $6 \cdot 10^{-4}$ both for the control u and the optimization variable q. For BFGS the indicators are slightly larger, but also decrease towards the end. If α is further reduced to 0.01, we observe one-step convergence for all three limited-memory methods, too, which can be explained by the fact that all but one time point are either active or zero then.

Next we consider the convergence of u and q in different norms. Table 5 displays the following errors for the last four steps of each optimization method:

$$e_{u,L^2}^k := \|u^k - \bar{u}\|_{L^2}, \qquad e_{u,H^1}^k := \|u^k - \bar{u}\|_{H^1}, \qquad e_{u,L^\infty}^k := \|u^k - \bar{u}\|_{L^\infty}, \tag{11}$$

and analogue definitions e_{q,L^2}^k, e_{q,H^1}^k, e_{q,L^∞}^k for q. The semismooth Newton method exhibits one-step convergence as soon as the active sets have converged. The other methods show a quick decrease of all values towards the relative tolerance during

	SN		Broyden		SR1		BFGS	
	r_u^k	r_q^k	r_u^k	r_q^k	r_u^k	r_q^k	r_u^k	r_q^k
L^2-norm	$3 \cdot 10^{-2}$	$2 \cdot 10^{-1}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$5 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$6 \cdot 10^{-2}$	$3 \cdot 10^{-2}$
	$3 \cdot 10^{-1}$	$7 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$1 \cdot 10^{-1}$	$2 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$4 \cdot 10^{-1}$	$3 \cdot 10^{-1}$
	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$6 \cdot 10^{-2}$	$1 \cdot 10^{-1}$
	$2 \cdot 10^{-10}$	$9 \cdot 10^{-11}$	$6 \cdot 10^{-3}$	$7 \cdot 10^{-3}$	$7 \cdot 10^{-4}$	$6 \cdot 10^{-4}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-2}$
L^∞-norm	$1 \cdot 10^{0}$	$3 \cdot 10^{0}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$5 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$7 \cdot 10^{-2}$	$7 \cdot 10^{-2}$
	$3 \cdot 10^{-1}$	$9 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$1 \cdot 10^{-1}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-1}$	$3 \cdot 10^{-1}$
	$4 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$5 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$
	$3 \cdot 10^{-10}$	$3 \cdot 10^{-10}$	$9 \cdot 10^{-3}$	$9 \cdot 10^{-3}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-2}$
H^1-semi-norm	$1 \cdot 10^{0}$	$7 \cdot 10^{-1}$	$5 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$6 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$8 \cdot 10^{-2}$	$1 \cdot 10^{-1}$
	$3 \cdot 10^{-1}$	$8 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$4 \cdot 10^{-1}$	$3 \cdot 10^{-1}$
	$6 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$
	$3 \cdot 10^{-10}$	$3 \cdot 10^{-9}$	$1 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$1 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$2 \cdot 10^{-2}$
$C^{0,1}$–semi-norm	$4 \cdot 10^{1}$	$1 \cdot 10^{1}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$7 \cdot 10^{-2}$	$2 \cdot 10^{-1}$
	$3 \cdot 10^{-1}$	$1 \cdot 10^{-1}$	$1 \cdot 10^{-1}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$3 \cdot 10^{-1}$	$3 \cdot 10^{-1}$
	$5 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$
	$4 \cdot 10^{-10}$	$8 \cdot 10^{-9}$	$9 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$5 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$2 \cdot 10^{-2}$

Table 4: Indicators of superlinear convergence with $\alpha = 0.1$, $\beta = 0.2$: The two columns per method show r_u^k and r_q^k for the four final iterations. Each group of four rows shows these indicators for different (semi-)norms.

the last four steps of the optimization run. In particular, SR1 yields the fastest reduction, while BFGS shows a significantly slower convergence here. As predicted by Theorem 2, part 2), we have $e^k_{u,L^2} \leq e^k_{q,L^2}$ and $e^k_{u,L^\infty} \leq e^k_{q,L^\infty}$ (since these inequalities are a consequence of the fact that $\operatorname{Prox}_{\varphi_1} : L^s \longrightarrow L^s$ is nonexpansive for any $s \in [1, \infty]$, they hold in all four methods under consideration). We observe that, in contrast, $e^k_{u,H^1} \leq e^k_{q,H^1}$ does not hold in general. However, the results indicate that e^k_{u,H^1} still goes to zero, which agrees with part 2) of Theorem 2 for the choice $Q = H^1(I)$ described in Lemma 7.

SN						Broyden					
e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}	e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}
$3 \cdot 10^{-2}$	$2 \cdot 10^{-1}$	$1 \cdot 10^{0}$	$7 \cdot 10^{-1}$	$1 \cdot 10^{0}$	$3 \cdot 10^{0}$	$3 \cdot 10^{-6}$	$5 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$6 \cdot 10^{-5}$	$9 \cdot 10^{-5}$	$9 \cdot 10^{-5}$
$9 \cdot 10^{-3}$	$2 \cdot 10^{-2}$	$4 \cdot 10^{-1}$	$5 \cdot 10^{-2}$	$3 \cdot 10^{-1}$	$3 \cdot 10^{-1}$	$3 \cdot 10^{-7}$	$6 \cdot 10^{-7}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$
$4 \cdot 10^{-4}$	$8 \cdot 10^{-4}$	$2 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-8}$	$3 \cdot 10^{-8}$	$7 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$4 \cdot 10^{-7}$	$4 \cdot 10^{-7}$
$6 \cdot 10^{-14}$	$7 \cdot 10^{-14}$	$7 \cdot 10^{-12}$	$7 \cdot 10^{-12}$	$4 \cdot 10^{-12}$	$4 \cdot 10^{-12}$	$9 \cdot 10^{-11}$	$2 \cdot 10^{-10}$	$7 \cdot 10^{-9}$	$2 \cdot 10^{-9}$	$4 \cdot 10^{-9}$	$4 \cdot 10^{-9}$
SR1						BFGS					
e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}	e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}
$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$4 \cdot 10^{-3}$	$6 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$8 \cdot 10^{-6}$	$8 \cdot 10^{-6}$
$2 \cdot 10^{-7}$	$5 \cdot 10^{-7}$	$2 \cdot 10^{-5}$	$7 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$3 \cdot 10^{-8}$	$3 \cdot 10^{-8}$	$3 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-6}$
$5 \cdot 10^{-9}$	$1 \cdot 10^{-8}$	$4 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$2 \cdot 10^{-7}$	$2 \cdot 10^{-7}$	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$	$1 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$8 \cdot 10^{-7}$	$8 \cdot 10^{-7}$
$4 \cdot 10^{-12}$	$7 \cdot 10^{-12}$	$4 \cdot 10^{-10}$	$3 \cdot 10^{-10}$	$2 \cdot 10^{-10}$	$2 \cdot 10^{-10}$	$1 \cdot 10^{-9}$	$3 \cdot 10^{-9}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$6 \cdot 10^{-8}$	$6 \cdot 10^{-8}$

Table 5: Errors in different norms for $\alpha = 0.1$, $\beta = 0.2$, for the final iterations.

6.2 Sparse control of the Bloch equation

As example for a nonconvex optimization problem we investigate the bilinear control of the Bloch equations in magnetic resonance imaging (without relaxation, in the rotating frame, and on-resonance). A realistic optimal control modeling for radiofrequency (RF) pulse design in slice-selective imaging is considered based on [21]. However, we add sparsity to the control model, which is a desirable feature in practice since the duty cycle of the RF amplifier is often limited. For details on magnetic resonance imaging we refer to [4]. As model problem we consider the slice-selective imaging with a single slice. Here, imaging data of a whole slice is to be acquired. The spatial field of view is described by its extent $\Omega \subset \mathbb{R}$ perpendicular to the slice direction. The slice itself is described by $\Omega_{\text{in}} \subset \Omega$ while the remaining part of Ω is denoted by $\Omega_{\text{out}} = \Omega \setminus \Omega_{\text{in}}$. The latter should not contribute to the data acquisition. The control problem is modeled as tracking of the nuclear magnetization vector $\mathbf{M} = \mathbf{M}(u) = (M_1, M_2, M_3)$ at the terminal time T. Specifically, we consider

$$\min_{u \in U_{\text{ad}}} \quad \hat{f}(u) + \frac{\alpha}{2}\|u\|^2_{L^2(I)} + \beta\|u\|_{L^1(I)} \qquad \text{s.t.} \tag{12}$$

$$\dot{\mathbf{M}}(t,x) = \gamma \mathbf{M}(t,x) \times \mathbf{B}(t,x) \quad \text{a.e. in } I \times \Omega, \qquad \mathbf{M}(0,x) = \mathbf{M}_0(x) \quad \text{a.e. in } \Omega \tag{13}$$

with $\alpha > 0$, proton gyromagnetic ratio $\gamma = 267.5380$ [rad/s/μT], given initial condition $\mathbf{M}_0(x)$, spatial domain $x \in \Omega = (-c, c)$ with $c = 0.06$ [m], and time $t \in I = (0, T)$ with $T = 2.69$ [ms]. The term $\hat{f}(u)$ is a tracking-type functional at the terminal time T describing the intended use of the RF pulse, see below. The external magnetic field $\mathbf{B}(t,x) = (u(t), v(t), w(t)x)$ depends on the RF pulse $(u, v) \in L^2(I)^2$ and the slice-selective gradient amplitude $w = w(t) \in L^2(I)$. While these three time-dependent functions can often be controlled, we consider for simplicity the situation in which $w \equiv 2$ is given and the RF pulses are real-valued, i.e., $v \equiv 0$. Hence, u is the control variable. Technical limitations of the RF amplifier are modeled as control constraints $U_{\text{ad}} = \{u \in L^2(I) : |u| \le u_{\max}\}$ with $u_{\max} = 1.2$ [$10^2\mu$T]. This value reflects a typical 3T magnetic resonance scanner hardware.

The specific example here is the optimization of a refocusing pulse, which is, among others, a central building block of the clinically important turbo spin echo based sequences. The initial condition results from assuming that an ideal 90°-excitation pulse for the same slice has been applied before, keeping the net magnetization vectors out of the slice in the steady state $(0,0,1)^T$ while exciting the slice itself. In particular, we set $\mathbf{M}_0 = \chi_{\Omega_{\text{out}}}(x)(0,0,1)^T + \chi_{\Omega_{\text{in}}}(x)(0,1,0)^T$. A slice of 1.65 [cm] thickness is assumed: $\Omega_{\text{in}} = [-0.00825, 0.00825]$. The aim of the refocusing is to flip the magnetization vectors in the $x-y$–plane in the interior Ω_{in} of the slice, which is modeled as rotation around the x axis with angle π. This desired magnetization pattern at the end time $t = T$ of the refocusing pulse is given by

$$\hat{f}(u) = \frac{1}{2}\int_{\Omega_{\text{in}}} (M_1(T,x))^2 + (M_2(T,x)+1)^2 dx + \frac{1}{2}\int_{\Omega_{\text{out}}} (1 - M_3(T,x))^2 dx,$$

recalling that $\mathbf{M} = \mathbf{M}(u)$. However, this tracking term for basic refocusing pulses is typically not used in numerical practice. Instead, we apply a more involved formulation of the desired state at the terminal time for advanced refocusing pulses, that we describe now. Because of practical reasons including robustness issues, refocusing pulses are generally applied within crusher gradients, cf. [4], which are additional sequence elements surrounding the RF pulse. These crusher gradients cannot be modeled by the depicted $\hat{f}(u)$. It seems that the only practical way to model tracking terms for refocusing pulses with ideal crusher gradients is to define them in the spin domain, cf. [4, 21]. Therefore, we choose an equidistant time grid $t_k = (k-1)\tau$, $k = 1, \dots, N_t$ with $N_t = 270$ points and step size $\tau = T/(N_t - 1) = 0.01$ [ms], together with piecewise constant w and control u with values $w_m, u_m, m = 1, \dots, N_t - 1$. This implies that the magnetic field $\mathbf{B}$ is piecewise constant. It is well-known that for piecewise constant magnetic field the Bloch equations (13) in a spatial point x_0 can be solved analytically as a sequence of rotations. This is expressed by using the Cayley–Klein parameters (a_m), $(b_m) \in \mathbb{C}$, $m = 1, \dots, N_t - 1$ with evolution

$$a_m = \alpha_m a_{m-1} - \beta_m^* b_{m-1}, \qquad b_m = \beta_m a_{m-1} + \alpha_m^* b_{m-1},$$

and with initial conditions $a_0 = 1$, $b_0 = 0$, cf. [18]. For the formula relating a_m, b_m and $\mathbf{M}(t_{m-1}, x_0)$ see [4, eq.(2.15)]. The coefficients α_m, β_m are given by

$$\alpha_m = \cos(\phi_m/2) + i\gamma\tau x_0 w_m \sin(\phi_m/2)/\phi_m, \qquad \beta_m = i\gamma\tau u_m \sin(\phi_m/2)/\phi_m,$$

with $\phi_m = -\gamma\tau\sqrt{u_m^2 + (x_0 w_m)^2}$. Since it is well-known that perfect refocusing with ideal crusher gradients is obtained through $|b(T,x)|^2 = \chi_{\Omega_{\text{in}}}(x)$ for a.e. $x \in \Omega$, the tracking term is given by

$$\hat{f}(u) = \frac{1}{2} \left\| \, |b(T,x)|^2 - \chi_{\Omega_{\text{in}}}(x) \, \right\|_{L^2(\Omega)}^2 . \tag{14}$$

Note that $b = b(u)$. In the numerical experiments we use $\hat{f}$ as defined in (14). The adjoint equation and the reduced gradient for this formulation are derived in the appendix of [21]. The spatial domain is discretized equidistantly in $N_x = 481$ points.

In accordance with the presentation in Section 4 we use a control-reduced problem formulation for (12–13). For the problem at hand this bears the advantage to iterate only on the small control vector with $N_t - 1 = 269$ entries, but not on the large state vector with $3N_xN_t = 389610$ entries. Consequently, B_k is a rather small matrix of format 269×269, and the linear algebra operations for its update and evaluation are cheap. In effect, the numerical effort is dominated clearly by the state and adjoint solves; concrete values are reported below.

We consider the same four optimization methods as in Section 6.1, i.e., a semismooth Newton method and the hybrid method with the Broyden, SR1 and BFGS update, respectively. If not mentioned otherwise, these methods are globalized with tr-cg. The stopping criterion is a relative tolerance of 10^{-5} for the residual norm $\|H\|_{L^2(I)}$. Unless declared otherwise, the following settings are applied for the hybrid methods: a limit of $L = 75$, $B_0 = 0$ for Broyden and SR1 methods, and $B_0 = \alpha I$ for BFGS. These initializations are selected because they turned out to be the most effective for the respective methods on this problem. Note that for Broyden and SR1 this is consistent with the numerical results for the optimal control of the heat equation in Section 6.1, cf. Table 1.

6.2.1 Comparison of the optimization methods

The optimization is initialized with a sinc-shaped RF pulse $f^0(t) = 1.8 \cdot \mathrm{sinc}(-2.2 + 4t/T)$. To maintain a good initial slice profile we use $q^0 = f^0 + \mathrm{sign}(f^0)\beta/\alpha$, which implies $f^0 = \sigma(q^0)$ and $u^0 = \Pi_{U_{\mathrm{ad}}}(\sigma(q^0)) = \Pi_{U_{\mathrm{ad}}}(f^0)$. The initialization, the corresponding slice profile $M_3(T, x)$, and the desired slice profile are depicted in the top of Figure 2. Also depicted are optimal controls for different α, β. The sparsity and bound properties of the solutions for different α, β are depicted in Table 6. If not mentioned otherwise, then we use $\alpha = 5 \cdot 10^{-4}$ and $\beta = 10^{-4}$ below. In all experiments we monitor that only runs leading to the same local minimizer are compared, which is important since the problem possesses several different minimizers.

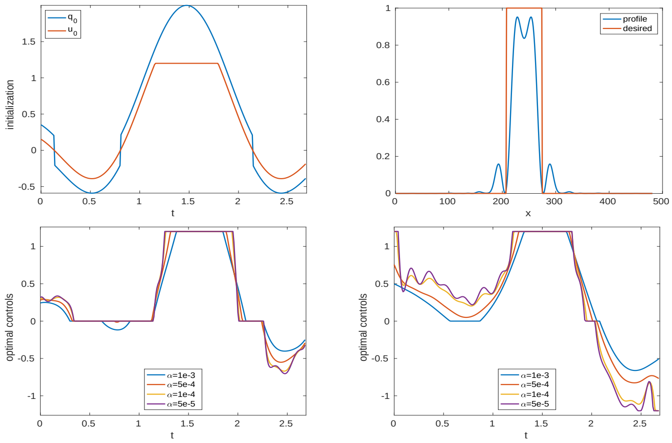

Fig. 2: Top: Initial q^0 and $u^0 = \Pi_{U_{\mathrm{ad}}}(\sigma(q^0))$ (left) and the corresponding slice profile $M_3(T, x)$ compared to the desired slice profile (right). Bottom: Optimal controls for different α with $\beta = 5 \cdot 10^{-4}$ (left) and $\beta = 5 \cdot 10^{-5}$ (right).

	$\lvert\mathcal{O}\rvert$					$\lvert\mathcal{A}\rvert$				
$\alpha \setminus \beta$	$5 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	10^{-4}	$5 \cdot 10^{-5}$	10^{-5}	$5 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	10^{-4}	$5 \cdot 10^{-5}$	10^{-5}
10^{-3}	73	58	37	31	11	47	47	44	43	42
$5 \cdot 10^{-4}$	93	76	20	5	1	57	57	54	51	50
10^{-4}	104	75	12	8	3	68	66	65	65	67
$5 \cdot 10^{-5}$	106	75	14	10	4	70	69	71	75	101

Table 6: Number of time instances with sparsity ($\lvert\mathcal{O}\rvert$), respectively, active box constraint ($\lvert\mathcal{A}\rvert$) out of 269 for different α (rows) and β (five columns each).

The first study compares the performance of four different semismooth Newton-type methods embedded in a trust-region cg framework for varying α, β and for different initializations of the quasi-Newton matrix B_0. First, the limited-memory methods are analyzed in Table 7 and compared to the semismooth Newton method. The up to four columns per method show the iteration counts for different B_0. The last row shows the mean value per column taken over the converged runs only. The runs that do not converge are marked with –. The first three columns display the parameters α, β and the iteration number of SN. The next two column groups show that the hybrid method with Broyden updates behaves quite similar to the variant with SR1 updates, the latter often requiring slightly fewer iterations. The choices $B_0 = 0$ and $B_0 = \alpha I$ yield fast convergence throughout all (α, β)-pairings, while the other two formulas turn out to be less efficient in this setting. Looking at the hybrid method with BFGS in the last column group we observe that its performance degenerates for large values of α. Apart from this surprising phenomenon the scaled identity yields good results also for BFGS. Using the best choice B_0 for each method, Broyden requires 41 iterations on average, SR1 35, and BFGS 83, compared to 16 for the semismooth Newton method. Since the latter has much more costly iterations due to the forward-backward solve of the second-order equations, it is important to compare the corresponding runtimes. They are included below, cf. Table 9.

α	β	SN	Broyden				SR1				BFGS		
			αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$	0	αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$	0	αI	$\frac{y^T s}{s^T s} I$	$\frac{y^T y}{y^T s} I$
10^{-3}	$3 \cdot 10^{-4}$	9	44	102	–	28	33	84	–	28	296	222	–
$5 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	11	29	97	–	38	31	143	–	27	137	143	–
10^{-4}	$3 \cdot 10^{-4}$	28	52	206	–	51	48	–	–	49	70	90	128
$5 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	32	64	243	–	61	52	–	–	61	67	76	149
10^{-3}	10^{-4}	9	43	70	–	23	27	71	–	23	–	122	–
$5 \cdot 10^{-4}$	10^{-4}	9	32	65	–	35	30	58	–	30	58	84	211
10^{-4}	10^{-4}	21	40	101	174	48	43	–	–	39	44	58	76
$5 \cdot 10^{-5}$	10^{-4}	30	51	145	–	56	49	–	–	54	54	63	89
10^{-3}	$5 \cdot 10^{-5}$	9	27	59	–	22	28	53	–	23	104	98	–
$5 \cdot 10^{-4}$	$5 \cdot 10^{-5}$	15	34	53	106	38	27	65	170	23	39	55	298
10^{-4}	$5 \cdot 10^{-5}$	16	44	104	135	50	40	131	–	41	41	46	71
$5 \cdot 10^{-5}$	$5 \cdot 10^{-5}$	21	49	115	284	54	46	265	–	49	49	46	90
10^{-3}	10^{-5}	10	27	52	–	22	20	66	174	19	218	83	–
$5 \cdot 10^{-4}$	10^{-5}	7	31	69	86	33	33	67	162	23	44	38	72
10^{-4}	10^{-5}	16	37	78	–	50	33	191	–	34	37	39	60
$5 \cdot 10^{-5}$	10^{-5}	15	46	124	168	46	45	223	–	42	49	52	78
	∅	16	41	105	159	41	38	109	127	35	87	120	83

Table 7: Iteration numbers of the trust-region hybrid quasi-Newton implementations with $L = 75$ for different α, β (rows) and B_0 (columns per method). The symbol – indicates that the relative tolerance is not met within 300 iterations. The last line depicts the mean iteration counts per column for the converged runs.

To address the choice of the limit parameter L we compare the performance of the hybrid methods for different limits in Table 8 based on the iteration counts. Depicted are four columns per method which differ in the choice of the limit ranging from $L = 25$ to $L = 100$. The rows show results for different α while keeping $\beta = 10^{-4}$

fixed. We stress that in combination with tr-cg it is appropriate to choose a limit L that is larger than typical values from the literature for globalization by line search methods. This is due to the fact that Steihaug-cg employs earlier breaks in the cg method leading to smaller and more steps. We observe that a limit of 25 is only adequate for Broyden and SR1 in the case of large $\alpha \geq 5 \cdot 10^{-4}$. For smaller α the limit should be increased to 50. In contrast, the performance of BFGS is less sensitive to the values of L that are investigated.

In the particular setting of optimal control problems with many state variables and few control variables, which is typically the case in practical applications of PDE-constrained optimization or Bloch-models, it pays off in runtime to use larger limits since this helps to save some iterations of the trust-region method while it increases the required time per trust-region iteration only marginally. This is due to the fact that the costs per trust-region iteration are largely dominated by the evaluation of objective and gradient. For example, the runtime of Broyden with $\alpha = 5 \cdot 10^{-5}$ for a limit of 75 (56 iterations) is 5.1 seconds, which is significantly lower than the 6.4 seconds that are needed with a limit of 50 (73 iterations). In both cases, around 90% of the runtime is spent on the evaluation of objective and gradient. Therefore, a limit of $L = 75$ is chosen for all subsequent studies. We emphasize that all runs converge to the same local minimizer independently of the limit parameter.

	Broyden				SR1				BFGS			
$\alpha \setminus L$	25	50	75	100	25	50	75	100	25	50	75	100
10^{-3}	23	23	23	23	23	23	23	23	–	–	–	–
$5 \cdot 10^{-4}$	61	35	35	35	69	30	30	30	54	58	58	58
10^{-4}	82	48	48	48	–	39	39	39	53	44	44	44
$5 \cdot 10^{-5}$	127	73	56	56	150	67	54	54	64	54	54	54

Table 8: Iterations of the hybrid quasi-Newton methods for different α and different limits L. The symbol – stands for not converging within 300 iterations.

To investigate mesh independence properties of the hybrid methods we perform runs with different temporal and spatial mesh sizes. The results are shown in Table 9 using iteration counts in the left table, respectively runtime (mean runtime in 5 runs, in seconds) in the right table. The rows depict results for different temporal refinements with $N_c = N_t - 1$ control points. The two columns per method show different spatial grids with $N_x = 481$, respectively, $N_x = 4811$ points. The finest example with $N_c = 17216$ and $N_x = 4811$ features 248 million degrees of freedom for the state variable. In all cases the same initial guess is used. Furthermore, the same local minimizer is attained in all runs, which allows for a direct comparison. The left table shows that the iteration counts of all four methods do not increase with N_c or N_x. In particular, SN, Broyden and SR1 exhibit nearly the same iteration count for any of the discretizations. The hybrid method with BFGS displays a constant iteration number per column with a reduced iteration count for the right column (larger N_x). Interestingly, its performance is rather similar to that of Broyden and SR1 for $N_x = 4811$, while for $N_x = 481$ it requires roughly twice as many iterations and twice as much runtime as Broyden and SR1.

The right table shows the mean values of five runtimes in seconds, measured for a single CPU without parallelization. Despite their higher iteration counts, the three hybrid methods have much smaller runtimes than the semismooth Newton method. The bottom line depicts the mean value per column of the runtime in microseconds divided by $N_x N_c$, a number that varies only slightly per column (and per row, although not displayed). We regard this quantity as an efficiency index and denote it by $\mathcal{E}$. In contrast, the runtime of the semismooth Newton method increases faster in N_c than linearly. We attribute this to the fact that the cg method requires more iterations for larger systems and that these iterations involve expensive operations for SN. Therefore, the speedup factor of the Broyden variant of the hybrid method over the semismooth Newton method rises with the number of time instances, starting at 7 and reaching 70 for $N_c = 17216$ and $N_x = 481$. Using SR1 updates leads to similar runtimes with a speedup of up to 68. As already seen in the left table, the use of BFGS updates produces higher iteration counts resulting in an increased runtime. Still, a speedup factor of up to 37 over the semismooth Newton method is reached.

	Iteration count								Runtime							
N_c	SN		Broyden		SR1		BFGS		SN		Broyden		SR1		BFGS	
269	10	10	32	34	35	27	89	41	23	181	3	33	3	26	8	39
538	9	9	28	34	29	27	73	38	46	507	7	69	5	52	13	73
1076	9	9	31	33	30	27	55	32	117	1193	13	139	11	114	20	137
2152	9	9	31	39	25	26	54	40	322	3199*	23	317	18	213	40	319
4304	9	9	29	28	29	26	53	36	1106*		43	438	45	407	77	559
8608	9	9	28	34	30	26	53	36	4485*		90	1092	95	841	168	1154
17216	9	9	28	28	27	30	53	36	12755*		183	2120*	188	1796*	341	2390*
∅	9.1	9.1	29.6	32.9	29.3	27	61.4	37		$\mathcal{E}$	23.2	26.1	21.2	20.6	44.0	28.5

Table 9: Iterations (left table) and runtime in seconds (right table) for different numbers of control points N_c (rows) and $N_x = 481, 4811$ (two columns per method). The runtimes are mean values from five runs, respectively, one run if marked with an asterisk. For the efficiency index $\mathcal{E}$ a smaller value indicates greater efficiency.

Let us take a closer look at the convergence properties of the different quasi-Newton updates in the trust-region method. To this end, the solution is first computed in high precision with the semismooth Newton method and a relative tolerance of 10^{-13}. Then the different optimization runs are performed with a relative tolerance of 10^{-7}, measured in $\|H(\cdot)\|_{L^2(I)}$. The results are displayed in Table 10, with the error norms defined as in the first example, see (11). The table shows that all methods reduce all six errors to approximately the size of the relative tolerance, but the semismooth Newton method needs much fewer iterations to achieve this, cf. Table 9. We attribute this to the strong nonconvexity of the bilinear problem at hand. As in the first example we have $e^k_{u,L^2} \leq e^k_{q,L^2}$ and $e^k_{u,L^\infty} \leq e^k_{q,L^\infty}$, but this relationship is not satisfied for the $H^1(I)$-norm, which, however, does not impede $e^k_{u,H^1} \to 0$.

SN						Broyden					
e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}	e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}
$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$4 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$7 \cdot 10^{-2}$	$7 \cdot 10^{-2}$	$6 \cdot 10^{-7}$	$9 \cdot 10^{-7}$	$2 \cdot 10^{-5}$	$6 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$
$6 \cdot 10^{-4}$	$9 \cdot 10^{-4}$	$2 \cdot 10^{-2}$	$6 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-2}$	$2 \cdot 10^{-7}$	$2 \cdot 10^{-7}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-6}$
$1 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$4 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$8 \cdot 10^{-8}$	$1 \cdot 10^{-7}$	$2 \cdot 10^{-6}$	$8 \cdot 10^{-7}$	$1 \cdot 10^{-6}$	$1 \cdot 10^{-6}$
$2 \cdot 10^{-9}$	$3 \cdot 10^{-9}$	$4 \cdot 10^{-8}$	$2 \cdot 10^{-8}$	$3 \cdot 10^{-8}$	$5 \cdot 10^{-8}$	$7 \cdot 10^{-9}$	$9 \cdot 10^{-9}$	$2 \cdot 10^{-7}$	$8 \cdot 10^{-8}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-7}$

SR1						BFGS					
e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}	e^k_{u,L^2}	e^k_{q,L^2}	e^k_{u,H^1}	e^k_{q,H^1}	e^k_{u,L^∞}	e^k_{q,L^∞}
$8 \cdot 10^{-7}$	$2 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$4 \cdot 10^{-5}$	$5 \cdot 10^{-8}$	$8 \cdot 10^{-8}$	$2 \cdot 10^{-6}$	$6 \cdot 10^{-7}$	$1 \cdot 10^{-6}$	$1 \cdot 10^{-6}$
$3 \cdot 10^{-7}$	$9 \cdot 10^{-7}$	$1 \cdot 10^{-5}$	$8 \cdot 10^{-6}$	$6 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$4 \cdot 10^{-8}$	$7 \cdot 10^{-8}$	$1 \cdot 10^{-6}$	$5 \cdot 10^{-7}$	$9 \cdot 10^{-7}$	$1 \cdot 10^{-6}$
$8 \cdot 10^{-8}$	$1 \cdot 10^{-7}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$1 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$4 \cdot 10^{-8}$	$6 \cdot 10^{-8}$	$1 \cdot 10^{-6}$	$4 \cdot 10^{-7}$	$7 \cdot 10^{-7}$	$9 \cdot 10^{-7}$
$1 \cdot 10^{-8}$	$3 \cdot 10^{-8}$	$2 \cdot 10^{-7}$	$3 \cdot 10^{-7}$	$2 \cdot 10^{-7}$	$6 \cdot 10^{-7}$	$3 \cdot 10^{-8}$	$5 \cdot 10^{-8}$	$1 \cdot 10^{-6}$	$3 \cdot 10^{-7}$	$6 \cdot 10^{-7}$	$8 \cdot 10^{-7}$

Table 10: Errors of the hybrid methods in different norms for the final iterations.

6.2.2 Comparison of the globalization techniques

This section is devoted to comparing the three globalizations tr-cg, ls-GMRES and nls-GMRES on the bilinear control problem of the Bloch equations. The globalizations are paired with the semismooth Newton method and the three limited-memory quasi-Newton methods. For each of these twelve combinations, 2000 optimization runs from a random initial (Matlab `rand`) q^0 are performed with the following parameters: $N_c = 269$, $N_x = 481$, $\alpha = 5 \cdot 10^{-4}, \beta = 10^{-4}$, up to 300 iterations, relative tolerance 10^{-4}, cg/GMRES tolerance 10^{-10}, up to 100 cg/GMRES iterations. The monotone line search operates on the residual $\|H(q)\|_{L^2(I)}$, while the non-monotone line search is tested with $N_{\text{ls}} = 2, 3, 4, 5$ based on the objective $f(u) + \varphi(u)$ and based on the residual $\|H(q)\|_{L^2(I)}$. Due to space limitations we show only the best results, which are obtained with $N_{\text{ls}} = 2$ and $R(u) = f(u) + \varphi(u)$.

Throughout the 8000 optimization runs with tr-cg, twelve different stationary points are observed, whose controls are depicted in Figure 3, divided into three sets (top row). Every set of four controls yields identical optimal values $\bar{m} := f(\bar{u}) + \varphi(\bar{u})$, control norms $\|\bar{u}\|_U$, and final magnetization (bottom row). The four controls are related by axial symmetry to the t-axis and the axis $t = T/2$. Since these four controls are equivalent in practical application, they are counted henceforth as one solution with multiplicity four. The relative occurrences of the resulting three candidates for a minimizer are depicted in the upper three rows of Table 11 in columns five to eight. The lower part of the table additionally displays the relative occurrences of runs that do not reach the prescribed relative tolerance within 300 iterations; they are labeled "not converged". The average objective value returned by the optimizer $\varnothing\ \bar{m}$ and the average runtime in seconds are also shown. In contrast to previous experiments they are taken over all runs here, i.e., they include also the "not converged" runs. In particular, we observe that SN, Br and SR always meet the relative tolerance. In contrast, one seventh of the BFGS runs does not converge. In fact, these runs yield the same twelve minimizers but fail to reach the prescribed tolerance, which is underlined by the agreeing values of $\varnothing\ \bar{m}$. This mean objective value is $\varnothing\ \bar{m} = 0.64 \cdot 10^{-3}$ for all four methods with tr-cg, which is significantly smaller than the

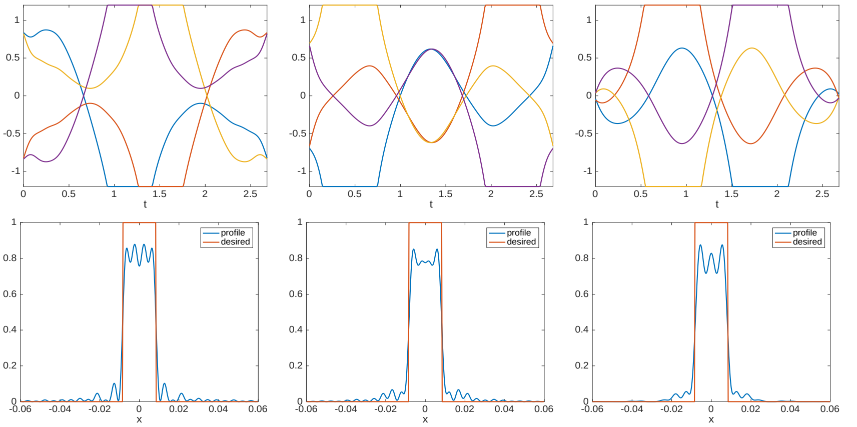

Fig. 3: Four optimized controls with the same objective value (top) and their slice profiles (bottom, desired state in red) for the three best observed candidates $\bar{m} \cdot 10^3 = 0.61906$ (left), $\bar{m} \cdot 10^3 = 0.64959$ (mid), and $\bar{m} \cdot 10^3 = 0.67280$ (right).

values achieved with the other globalization techniques. The mean runtime shows a clear speedup of the hybrid methods compared to SN, despite the fact that this is a small scale example with $N_t = 270$ and $N_x = 481$.

In contrast to the trust-region method, the line search globalizations find many more stationary points, 78 in total. However, 49 of these have a prohibitively high cost. They are summarized in lines number 10 and 11 of Table 11. The results with monotone line search (ls-GMRES) are depicted in the fourth column group of Table 11. Semismooth Newton with a basic monotone line search on the residual $\|H(q)\|_{L^2(I)}$ quickly converges to a noncompetitive minimizer in nearly all cases. The three quasi-Newton methods yield smaller cost values in the mean, but most of the runs fail to match the prescribed relative tolerance. The mean optimal values of all four methods are much larger than those obtained with tr-cg.

The non-monotone line search nls-GMRES is more effective than ls-GMRES for all three quasi-Newton methods, see the last column group of Table 11. We note that the number of runs that do not converge is smaller than for the monotone line search. In particular, Broyden and BFGS converge in most of the cases. Moreover, the best control and the top three controls are found more often leading to much better average optimal values compared to ls-GMRES. However, excellent values similar to those of tr-cg are attained only for BFGS. Notably, the semismooth Newton method does not benefit from the non-monotone line search; it behaves similarly as with ls-GMRES. It is also worth mentioning that the average runtimes of the tr-cg hybrid quasi-Newton methods are only slightly above those of the nls-GMRES variant.

Summarizing, in this application problem the tr-cg globalization robustly delivers the top-three candidates for all four optimization methods. In contrast, the line search globalizations often have difficulties with convergence for the quasi-Newton methods, and tend to noncompetitive solutions for the semismooth Newton method. Thus, for optimal control of the Bloch equations tr-cg should clearly be preferred over a line search globalization for the semismooth Newton, the hybrid Broyden, and the SR1 method. For BFGS, both tr-cg and nls-GMRES work equally well.

				% (tr-cg)				% (ls-GMRES)				% (nls-GMRES)			
nr	mult	$\bar{m} \cdot 10^3$	$\|\bar{u}\|_{L^2}$	SN	Br	SR	BF	SN	Br	SR	BF	SN	Br	SR	BF
1	4	0.61906	1.1961	41.6	36.7	38.4	37.3		11.1	0.2	7.7	0.2	39.6	18.8	44.8
2	4	0.64959	1.1367	45.2	43.8	43.7	40.9		3.8	0.2	1.9	0.1	20.9	7.8	31.7
3	4	0.67280	1.1215	13.2	19.6	17.9	7.7		1.1	0.1	1.0		6.7	4.3	6.6
4	2	0.67608	1.1020									0.1		0.1	
5	3	0.69636	1.1488					0.1	0.2			0.1			
6	3	1.11919	1.1522						0.1				0.1	0.5	
7	1	1.57083	1.0094									0.1		0.1	
8	4	1.61472	1.0902						0.1			0.1		0.2	
9	4	1.82260	1.6886										1.4	0.3	0.2
10	36	$\in [2,3]$						0.1	4.2	0.3	1.9		14.6	2.8	4.8
11	13	> 3						99.2				88.4			
			not converged				14.2	0.7	79.5	99.2	87.6	11.1	16.9	65.3	11.9
			$\varnothing\ \bar{m} \cdot 10^3$	0.64	0.64	0.64	0.64	4.18	2.78	2.78	2.78	4.14	0.94	1.17	0.72
			$\varnothing$ runtime	23	4	3	6	19	3	2	3	19	3	4	5

Table 11: Quality of the solutions for different optimization and globalization methods for $\alpha = 5 \cdot 10^{-4}, \beta = 10^{-4}$. 2000 optimization runs from random initializations are performed for each combination. Depicted are the multiplicity of the minimizer, the optimal value $\bar{m}$, and the norm of the optimal control. The next three column groups show the relative occurrence (%) of the respective solution, separately for tr-cg, ls-GMRES and nls-GMRES. Each column group is divided into the four methods semismooth Newton (SN), Broyden (Br), SR1 (SR), and BFGS (BF). The three bottom lines depict the percentage of runs that did not converge, the average objective value $\varnothing\ \bar{m}$, and the average runtime, both averaged over all runs.

7 Conclusions

In this paper we have studied a hybrid approach for nonsmooth optimal control problems that blends semismooth Newton and quasi-Newton methods. We established its local superlinear convergence and provided numerical results to show that it has significantly lower runtime than semismooth Newton methods. A matrix-free limited-memory truncated trust-region variant seems to be particularly promising.

Acknowledgements This work was supported in part by the Austrian Science Fund (FWF) in the context of the "SFB F32-N18" (Mathematical Optimization and Applications in Biomedical Sciences) in the projects F3201 and F3202.

A Trust-region globalization of the hybrid method

We state the precise algorithm of tr-cg that is employed in the numerical experiments. It is designed for solving (P) from Section 3 and uses the notation of that section. The objective of (P) is denoted by $J : U \to \mathbb{R}$, i.e., $J(u) := f(u) + \varphi(u)$. For the norm $\|\cdot\|$ that appears in the algorithm we used $\|\cdot\|_U$, which worked well. The mapping H is given by $H(q) = \nabla\hat{f}(\mathrm{Prox}_{\varphi_\gamma}(q)) + \gamma q$, cf. Lemma 1.

Algorithm 2: Hybrid semismooth quasi-Newton-cg method with trust-region globalization

```
Input: $0 < \mathrm{tol}_{\mathrm{tr}}, \mathrm{tol}_{\mathrm{cg}} \ll 1$;  $\mathrm{maxit}_{\mathrm{tr}}, \mathrm{maxit}_{\mathrm{cg}} \in \mathbb{N}$;  initial guess $(u^0, B_0)$;
       trust-region parameters $0 < \varrho_0 \le \varrho_{\max}$,  $0 < \sigma_1 < \sigma_2 < \sigma_3 < 1$,  $0 < f_1, f_3 < 1 < f_2$
Set $k = 0, \varrho = \varrho_0$;  compute $q^0 := -\frac{1}{\gamma}\nabla\hat{f}(u^0)$;  compute $H(q^0)$;  choose $M \in \partial\mathrm{Prox}_{\varphi_\gamma}(q^0)$
Define $\langle x, y\rangle = (x, My)_U$
while $\left[\|H(q^k)\| > \mathrm{tol}_{\mathrm{tr}}\|H(q^0)\|\ \ and\ \ k \le \mathrm{maxit}_{\mathrm{tr}}\right]$ do        // trust-region loop
    Set $p^0 = r^0 = -H(q^k)$, $\delta q = 0$, $i = 0$
    while $\left[\|r^i\| > \mathrm{tol}_{\mathrm{cg}}\|r^0\|\ \ and\ \ i \le \mathrm{maxit}_{\mathrm{cg}}\right]$ do        // Steihaug-cg loop
        Set $\tilde{M} = B_k M + \gamma I$;  compute $\tilde{M}p^i$
        if $\langle p^i, \tilde{M}p^i\rangle \le 0$ then        // negative curvature
            Compute $\max\{\tau : \|\delta q + \tau p^i\| \le \varrho\}$        // go to boundary of trust-region
            Set $\delta q = \delta q + \tau p^i$
            break
        end
        Compute $\alpha = \|r^i\|/\langle p^i, \tilde{M}p^i\rangle$
        if $\|\delta q + \alpha p^i\| \ge \varrho$ then        // step too large
            Compute $\max\{\tau : \|\delta q + \tau p^i\| \le \varrho\}$        // go to boundary of trust-region
            Set $\delta q = \delta q + \tau p^i$
            break
        end
        Set $r^{i+1} = r^i - \alpha\tilde{M}p^i$
        Set $p^{i+1} = r^{i+1} + \|r^{i+1}\|^2/\|r^i\|^2 p^i$
        Set $\delta q = \delta q + \alpha p^i$,   $i = i + 1$
    end
    Compute $\delta J_a = J(\mathrm{Prox}_{\varphi_\gamma}(q^k)) - J(\mathrm{Prox}_{\varphi_\gamma}(q^k + \delta q))$        // actual decrease
    Compute $\delta J_m = -\frac{1}{2}\langle\delta q, \tilde{M}\delta q\rangle - \langle\delta q, H(q^k)\rangle$        // predicted decrease
    if $\left[\delta J_a > \varepsilon\ \ and\ \ \delta J_a > \sigma_1\delta J_m\right]$ then        // accept step
        Set $q^{k+1} = q^k + \delta q$
        Compute $H(q^{k+1})$;  choose $M \in \partial\mathrm{Prox}_{\varphi_\gamma}(q^{k+1})$
        Define $\langle x, y\rangle = (x, My)_U$
        if $|\delta J_a/\delta J_m - 1| \le 1 - \sigma_3$ then        // increase radius
            Set $\varrho = \min\{f_2\varrho, \varrho_{\max}\}$
        else if $|\delta J_a/\delta J_m - 1| > 1 - \sigma_2$ then        // decrease radius
            Set $\varrho = f_3\varrho$
        end
    else
        Set $\varrho = f_1\varrho$, $q^{k+1} = q^k$        // decrease radius
    end
    Set $s_u^k = \mathrm{Prox}_{\varphi_\gamma}(q^{k+1}) - \mathrm{Prox}_{\varphi_\gamma}(q^k)$, $y^k = \nabla\hat{f}(\mathrm{Prox}_{\varphi_\gamma}(q^{k+1})) - \nabla\hat{f}(\mathrm{Prox}_{\varphi_\gamma}(q^k))$
    Compute $B_{k+1}$ by quasi-Newton update
    Set $k = k + 1$
end
Output: $(q^k, \mathrm{Prox}_{\varphi_\gamma}(q^k))$
```

References

1. Adly, S., Ngai, H.V.: Quasi-Newton methods for solving nonsmooth equations: Generalized Dennis-Moré theorem and Broyden's update. J. Convex Anal. **25**(4), 1075–1104 (2018)
2. Appell, J., Zabrejko, P.P.: Nonlinear Superposition Operators. Cambridge Tracts in Mathematics. Cambridge University Press (1990). DOI 10.1017/CBO9780511897450
3. Bauschke, H.H., Combettes, P.L.: Convex analysis and monotone operator theory in Hilbert spaces, 2nd edn. Springer (2017). DOI 10.1007/978-3-319-48311-5
4. Bernstein, M.A., King, K.F., Zhou, X.J.: Handbook of MRI Pulse Sequences. Elsevier Academic Press (2004). DOI 10.1016/B978-012092861-3/50003-0
5. Borzì, A., Schulz, V.: Computational optimization of systems governed by partial differential equations, vol. 8. SIAM (2012). DOI 10.1137/1.9781611972054
6. Byrd, R.H., Nocedal, J., Schnabel, R.B.: Representations of quasi-newton matrices and their use in limited memory methods. Math. Program. **63**(1 (B)), 129–156 (1994). DOI 10.1007/BF01582063
7. Chipot, M.: Elements of nonlinear analysis. Birkhäuser (2000). DOI 10.1007/978-3-0348-8428-0
8. De los Reyes, J.C.: Numerical PDE-constrained optimization. Springer (2015). DOI 10.1007/978-3-319-13395-9
9. Griewank, A.: The local convergence of broyden-like methods on lipschitzian problems in hilbert spaces. SIAM J. Numer. Anal. **24**(3), 684–705 (1987). DOI 10.1137/0724045
10. Grissom, W.A., Setsompop, K., Hurley, S.A., Tsao, J., Velikina, J.V., Samsonov, A.A.: Advancing RF pulse design using an open-competition format: Report from the 2015 ISMRM challenge. Magnetic Resonance in Medicine **78**(4), 1352–1361 (2017). DOI 10.1002/mrm.26512
11. Hinze, M., Kunisch, K.: Second order methods for optimal control of time-dependent fluid flow. SIAM J. Control Optim. **40**(3), 925–946 (2001). DOI 10.1137/S0363012999361810
12. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE constraints, *Mathematical Modelling: Theory and Applications*, vol. 23. Springer (2009). DOI 10.1007/978-1-4020-8839-1
13. Ito, K., Kunisch, K.: Lagrange Multiplier Approach to Variational Problems and Applications. SIAM (2008). DOI 10.1137/1.9780898718614
14. Kinderlehrer, D., Stampacchia, G.: An introduction to variational inequalities and their applications, reprint of the 1980 original edn. SIAM (2000). DOI 10.1137/1.9780898719451
15. Mannel, F., Rund, A.: A hybrid semismooth quasi-Newton method for structured nonsmooth operator equations in Banach spaces. Submitted (2019); preprint: https://imsc.uni-graz.at/mannel/sqn1.pdf
16. Muoi, P.Q., Hào, D.N., Maass, P., Pidcock, M.: Semismooth Newton and quasi-Newton methods in weighted ℓ^1-regularization. J. Inverse Ill-Posed Probl. **21**(5), 665–693 (2013). DOI 10.1515/jip-2013-0031
17. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer Series in Operations Research and Financial Engineering. Springer (2006). DOI 10.1007/978-0-387-40065-5
18. Pauly, J., Le Roux, P., Nishimura, D., Macovski, A.: Parameter relations for the Shinnar–Le Roux selective excitation pulse design algorithm. IEEE Transactions on Medical Imaging **10**(1), 53–65 (1991). DOI 10.1109/42.75611
19. Pieper, K.: Finite element discretization and efficient numerical solution of elliptic and parabolic sparse control problems. PhD Thesis, Technische Universität München, Munich (2015). URL `https://nbn-resolving.de/urn/resolver.pl?nbn:de:bvb:91-diss-20150420-1241413-1-4`
20. Robinson, S.M.: Normal maps induced by linear transformations. Math. Oper. Res. **17**(3), 691–714 (1992). DOI 10.1287/moor.17.3.691
21. Rund, A., Aigner, C., Kunisch, K., Stollberger, R.: Magnetic resonance RF pulse design by optimal control with physical constraints. IEEE Transactions on Medical Imaging **37**(2), 461–472 (2018). DOI 10.1109/TMI.2017.2758391
22. Rund, A., Aigner, C.S., Kunisch, K., Stollberger, R.: Simultaneous multislice refocusing via time optimal control. Magnetic Resonance in Medicine **80**(4), 1416–1428 (2018). DOI 10.1002/mrm.27124
23. Sachs, E.: Convergence rates of quasi-Newton algorithms for some nonsmooth optimization problems. SIAM J. Control Optim. **23**, 401–418 (1985). DOI 10.1137/0323026
24. Schiela, A.: A simplified approach to semismooth Newton methods in function space. SIAM J. Optim. **19**(3), 1417–1432 (2008). DOI 10.1137/060674375
25. Steihaug, T.: The conjugate gradient method and trust regions in large scale optimization. SIAM J. Numer. Anal. **20**, 626–637 (1983). DOI 10.1137/0720042
26. Tröltzsch, F.: Optimal control of partial differential equations. Theory, methods and applications, vol. 112. AMS (2010). DOI 10.1090/gsm/112
27. Ulbrich, M.: Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces. MOS-SIAM Series on Optimization. SIAM (2011). DOI 10.1137/1.9781611970692