Optimization with learning-informed differential equation constraints and its applications

Guozhi Dong[1,2], Michael Hintermüller[1,2], and Kostas Papafitsoros[2]

[1]Institute for Mathematics, Humboldt University of Berlin, Unter den Linden 6, 10099 Berlin, Germany
[2]Weierstrass Institute for Applied Analysis and Stochastics, Mohrenstrasse 39, 10117 Berlin, Germany

Abstract

Inspired by applications in optimal control of semilinear elliptic partial differential equations and physics-integrated imaging, differential equation constrained optimization problems with constituents that are only accessible through data-driven techniques are studied. A particular focus is on the analysis and on numerical methods for problems with machine-learned components. For a rather general context, an error analysis is provided, and particular properties resulting from artificial neural network based approximations are addressed. Moreover, for each of the two inspiring applications analytical details are presented and numerical results are provided.

AMS subject classifications. 49M15, 65J15, 65J20, 65K10, 90C30, 35J61, 68T07

Keywords. Artificial neural network; optimal control; semilinear PDEs; integrated physics-based imaging; learning-informed model; quantitative MRI; semi-smooth Newton SQP algorithm

1 Introduction

Consider the optimization problem

$$\text{minimize } J(y,u) := \frac{1}{2}\left\|Ay - g\right\|_H^2 + \frac{\alpha}{2}\left\|u\right\|_U^2, \quad \text{over } (y,u) \in Y \times U,$$
$$\text{subject to (s.t.) } e(y,u) = 0, \text{ and } u \in \mathcal{C}_{ad}, \tag{1.1}$$

where $y \in Y$, $u \in U$ are the state and control variables, respectively, with Y a suitable Banach space and U a Hilbert space. Moreover, $g \in H$ denotes given data with H the pertinent Hilbert space, $\alpha > 0$ is the control cost, and $A : Y \to H$ is a bounded linear (observation) operator, i.e., $A \in \mathcal{L}(Y, H)$. While in (1.1) feasible controls u are confined to a nonempty, closed, and convex set $\mathcal{C}_{ad}$, the relationship between admissible controls and states is through the equality constraint associated with a possibly nonlinear operator $e : Y \times U \to Z$, with Z a Banach space. Often, $e(y, u) = 0$ is given by (a system of) ordinary or partial differential equations (ODEs or PDEs) describing, e.g., underlying physics. For the ease of discussion we assume that, for given $u \in U$, there is a unique $y \in Y$ such that $e(y, u) = 0$. This allows us to write

$$y = \Pi(u),$$

where Π denotes the (implicitly defined) control-to-state map with $e(\Pi(u), u) = 0$. Given Π, a popular approach in the study of (1.1) is based on the reduced problem

$$\text{minimize } \mathcal{J}(u) := \frac{1}{2}\left\|Q(u) - g\right\|_H^2 + \frac{\alpha}{2}\left\|u\right\|_U^2, \quad \text{over } u \in U,$$
$$\text{s.t. } u \in \mathcal{C}_{ad}, \tag{1.2}$$

where $Q := A\Pi(\cdot) : U \to H$. Note that $\mathcal{J}(u) = J(\Pi(u), u)$.

In general, (1.1) or its reduced form (1.2) represent a class of optimal control problems, for which a plethora of studies exist in the literature; see, e.g., [44] for an introduction and [19, 29, 35] as well as the references therein for more details. In contrast, in many applications one is confronted with control problems where e or, alternatively, Π are only partly known along with measurement data which can be exploited to obtain (approximations) of missing information. Such minimization tasks have barely been treated in the literature and motive the present work. In order to inspire such a setting, we briefly highlight here two classes of applications which will be further studied from Section 4 onwards.

Our first motivating example is related to the fact that many phenomena in engineering, physics or life sciences, for instance, can be modeled by elliptic partial differential equations of the form

$$\left.\begin{array}{ll} Ly + f(x,y) = Ru & \text{in } \Omega, \\ b(x)\partial_\nu y + d(x)y = 0 & \text{on } \partial\Omega. \end{array}\right\} \tag{1.3}$$

Here L denotes a second-order linear elliptic partial differential operator with measurable, bounded and symmetric coefficients, $f(x,y)$ is a nonlinearity, and R models the impact of the control action u. Moreover, b and d are given coefficient functions. The set $\Omega \subset \mathbb{R}^d$ represents the underlying domain with boundary $\partial\Omega$, and ∂_ν denotes the derivative along the outward (unit) normal ν to Ω. Often the precise form of f is unknown, but rather only accessible through a data set $D := \{(y_i, u_i) : e(y_i, u_i) \approx 0, i = 1, \ldots, n_D\}$, $n_D \in \mathbb{N}$, i.e., given pre-specified control actions, one collects associated state responses (through measurements or computations). Utilizing data-driven approximation techniques such as artificial neural networks (ANNs), one may then get access to a data-driven model of f which can be used even outside the range of the data set D to yield a valid model of the underlying real-world process. In such a setting, associated optimal control problems depend on approximations $\mathcal{N}$ of f, and theoretical investigations as well as numerical solutions of the control problem need to take the construction of $\mathcal{N}$ into account.

The second example comes from quantitative magnetic resonance imaging - qMRI. In this context, one integrates a mathematical model of the acquisition physics (the Bloch equations [16]) into the associated image reconstruction task in order to relate qualitative information (such as the net magnetization $y = \rho m$) with objective, tissue dependent quantitative information (such as T_1 and T_2, the longitudinal and the transverse relaxation times, respectively, or the proton spin density ρ). This model is then used to obtain quantitative reconstructions from subsampled measurement data g in k-space by a variational approach. The provision of such quantitative reconstructions is highly important, e.g., for subsequent automated image classification procedures to identify tissue anomalies. Moreover, in [16] it is demonstrated that such an *integrated physics-based* approach is superior to the state-of-the-art technique of magnetic resonance fingerprinting (MRF) [32] and its improved variants [14, 34]. Specifically in MRI, acquisition data are obtained at different pre-specified times (read-out times) $t_1, \ldots, t_L$, during which the magnetization of the matter is excited through the control of a time dependent external magnetic field B. Given $u = (T_1, T_2, \rho)$, the magnetization time vector at $t_1, \ldots t_L$ is then given by $y = \Pi(u)$, where Π denotes the solution map associated with a discrete version of the Bloch equations. Crucial to this approach is the fact that, at least for specific variations of the external magnetic field B, explicit formulas for the solution map of the Bloch equations are available. For instance, in [14] and [16] Inversion Recovery balanced Steady-State Free Precession (IR-bSSFP) [41] is used which involves certain flip angle sequence patterns that characterize the external magnetic field B. These flip angle patterns allow for a simple approximation of the solutions of the Bloch equations at the read-out times through a recurrence formula. However, in general, it is quite typical that for more complicated external magnetic fields one does not have at hand explicit representations for the Bloch solution map. More generally, for most nonlinear differential equations (including those relevant in image reconstruction tasks) explicit solution maps might be too complicated to obtain. However, one may employ numerical methods to approximate their solutions $(y_i)_{i=1}^{n_D}$ given a specific (coarse) selection of parameters $(u_i)_{i=1}^{n_D}$ within a certain range. This generates a data set D which is then employed in a learning procedure to generate an ANN based approximation $\Pi_\mathcal{N}$ of Π. This gives rise to $Q_\mathcal{N} := A\Pi_\mathcal{N}$ in (1.2) and requires an associated analytical as well as numerical treatment of the (reduced) minimization problem.

In general, learning-informed models are getting increasingly more popular in different scientific fields nowadays. Some works focus on the design of ANNs, e.g., by constructing novel network architectures [7], or on developing fast and reliable algorithms in order to train ANNs more efficiently [10]. More relevantly for our present work, ANNs have been applied to the simulation of differential dynamical systems [39] and high dimensional partial differential equations [22, 42], as well as to the coefficient estimation in nonlinear partial differential equations [30], also in connection with optimal control [17, 21] and inverse

problems [5]. Note, however, that in our approach neural networks do not aim to approximate the solution of (1.1), but rather they are part of the physical process encoded in Π. We emphasize that this is a different strategy to some of the recent works [3, 8] in the literature that focus on learning the entire model or reconstruction process. More precisely, in the present work we suggest to use an operator $\Pi_{\mathcal{N}}$ that is induced by trained neural networks modelling the equality constraint (with, e.g., f replaced by an ANN-based model $\mathcal{N}$ in our example (1.3)) or its (implicitly defined) solution map Π. In such a setting, existence, convergence, stability and error bounds of the corresponding approximations need to be analyzed. Particularly, we are interested in the requirements on the mathematical settings of both neural networks and differential equations for the robustness of the proposed framework, and the error propagation from the neural network approximation to the solution of the optimal control problem. Moreover, in the case of partial differential equations, when replacing f by $\mathcal{N}$, the regularity of solutions has to be checked carefully before approaching the optimal control problem. Further, from a numerical viewpoint, in order to use derivative-based numerical methods, it is important for these approximating solution maps to have certain smoothness. This aspect is typically tied to the regularity of the activation functions employed in ANN approximations.

The remaining part of the paper is organized as follows: Section 2 provides a general error analysis for solutions of the proposed learning-informed framework. Some basic definitions and approximation properties of artificial neural networks are recalled in Section 3, and Section 4 presents a concrete case study on optimal control of semilinear elliptic equations with general nonlinearities, including both error analysis and numerical results. Section 5 contains another case study on quantitative magnetic resonance imaging, again including computational results.

2   Mathematical analysis of the general framework problem

We start our analysis by studying (1.2) or its variant where Q, the original physics-based operator, is replaced by a (data-driven) approximation. Existence of a solution to (1.2) follows from standard arguments which are provided here for the sake of completeness.

Proposition 2.1. *Suppose that Q is weakly-weakly sequentially closed, i.e., if $u_n \xrightarrow{U} u$ and $Q(u_n) \xrightarrow{H} \bar{g}$, then $\bar{g} = Q(u)$. Then (1.2) admits a solution $\bar{u} \in U$. In the special case where $\mathcal{C}_{ad}$ is a bounded set of a subspace $\hat{U}$ which is compactly embedded into U, it suffices that Q is strongly-weakly sequentially closed to guarantee existence of a solution to (1.2).*

Proof. Suppose that Q is weakly-weakly sequentially closed and let $(u_n)_{n \in \mathbb{N}} \subset \mathcal{C}_{ad}$ be an infimizing sequence for (1.2). Since $\alpha > 0$, $(u_n)_{n \in \mathbb{N}}$ is bounded in U, and thus we can extract an (unrelabelled) weakly convergent subsequence, i.e., $u_n \xrightarrow{U} \bar{u}$ for some $\bar{u} \in U$. Since $\mathcal{C}_{ad}$ is strongly closed and convex, it is weakly closed and therefore $\bar{u} \in \mathcal{C}_{ad}$. Moreover, since the sequence $(Q(u_n))_{n \in \mathbb{N}}$ is also bounded in Y, passing to a subsequence if necessary, we get that there exists a $\bar{g} \in H$ such that $Q(u_n) \xrightarrow{H} \bar{g}$. Due to the weak sequential closedness we have $\bar{g} = Q(\bar{u})$. Finally, from the weak lower semicontinuity of $\| \cdot \|_H$ and $\| \cdot \|_U$ we have $\mathcal{J}(\bar{u}) \leq \liminf_{n \to \infty} \mathcal{J}(u_n) = \inf_{u \in \mathcal{C}_{ad}} \mathcal{J}(u)$ and hence $\bar{u}$ is a solution of (1.2).

For the special case let $(u_n)_{n \in \mathbb{N}}$ again be an infimizing sequence for (1.2). Due to the compact embedding, we have that $(u_n)_{n \in \mathbb{N}}$ has an (unrelabelled) subsequence such that $u_n \to \bar{u}$ strongly in U as $n \to \infty$. Then the proof follows the same steps as above. $\square$

Remark 2.2. *We note here that in many examples in optimal control of (semilinear) PDEs, the control-to-state map actually maps U to a solution space Y which is of higher regularity than H and even compactly embeds into it; e.g., $Y := H^1(\Omega) \hookrightarrow L^2(\Omega) =: H$. Provided that the control-to-state map is bounded, in that case weak convergence in U results, up to subsequences, in strong convergence in H with the latter used to show closedness of the control-to-state operator.*

Assuming that Q is Fréchet differentiable with derivative $Q'(\cdot) \in \mathcal{L}(U, H)$, the first-order optimality condition of (1.2) is

$$\langle \mathcal{J}'(\bar{u}), u - \bar{u} \rangle_{U^*, U} \geq 0 \quad \text{for all } u \in \mathcal{C}_{ad}, \tag{2.1}$$

where $\mathcal{J}'(\bar{u}) \in \mathcal{L}(U, \mathbb{R}) =: U^*$ is the Fréchet derivative of $\mathcal{J}$ at $\bar{u}$, and $\langle \cdot, \cdot \rangle_{U^*, U}$ denotes the duality pairing between U and its dual U^*. Utilizing the structure of $\mathcal{J}$ we get

$$\left\langle (Q'(\bar{u}))^* \iota_H^{-1}(Q(\bar{u}) - g) + \alpha \iota_U^{-1} \bar{u}, u - \bar{u} \right\rangle_{U^*, U} \geq 0 \quad \text{for all } u \in \mathcal{C}_{ad},$$

or alternatively

$$\bar{u} = \mathcal{P}_{\mathcal{C}_{ad}} \left(-\frac{\iota_U (Q'(\bar{u}))^* \iota_H^{-1} (Q(\bar{u}) - g)}{\alpha} \right),$$

where $\mathcal{P}_{\mathcal{C}_{ad}}$ is the projection in U onto $\mathcal{C}_{ad}$, and $\iota_H : H^* \to H$ as well as $\iota_U : U^* \to U$ are Riesz isomorphisms, respectively. For ease of notation, however, we will leave off the Riesz maps in what follows whenever there is no confusion.

We now proceed to the error analysis of (1.2), where we assume that $(Q_n)_{n \in \mathbb{N}}$ is a family of operators approximating Q, and clarify the convergence of the associated minimizers $u_n \in \mathcal{C}_{ad}$.

Theorem 2.3. *Let Q and Q_n, $n \in \mathbb{N}$, be weakly sequentially closed operators with*

$$\|Q(u) - Q_n(u)\|_H \leq \epsilon_n, \quad \text{for all } u \in \mathcal{C}_{ad}, \tag{2.2}$$

and $\epsilon_n \downarrow 0$. Furthermore let $(u_n)_{n \in \mathbb{N}}$ be a sequence of minimizers of (1.2) with Q replaced by Q_n for all $n \in \mathbb{N}$. Then, we have the strong convergences

$$u_n \to \bar{u} \text{ in } U, \quad \text{and} \quad Q_n(u_n) \to Q(\bar{u}) \text{ in } H, \quad \text{as } n \to \infty, \tag{2.3}$$

where $\bar{u}$ is a minimizer of (1.2).

Proof. As $(u_n)_{n \in \mathbb{N}}$ is a sequence of minimizers, we have for $C := \max_n \epsilon_n < \infty$ and every $u \in \mathcal{C}_{ad}$:

$$\frac{1}{2} \|Q_n(u_n) - g\|_H^2 + \frac{\alpha}{2} \|u_n\|_U^2 \leq \|Q(u) - g\|_H^2 + C^2 + \frac{\alpha}{2} \|u\|_U^2.$$

Note also that $\|Q(u_n)\|_H \leq \|Q_n(u_n)\|_H + \epsilon_n$. Hence $(u_n)_{n \in \mathbb{N}}$, $(Q(u_n))_{n \in \mathbb{N}}$ and $(Q_n(u_n))_{n \in \mathbb{N}}$ are bounded sequences and therefore there exist (unrelabelled) subsequences and $\bar{u} \in U$ such that $u_n \xrightarrow{U} \bar{u}$ with $\bar{u} \in \mathcal{C}_{ad}$ by weak closedness, $Q(u_n) \xrightarrow{H} Q(\bar{u})$, and $Q_n(u_n) \xrightarrow{H} Q(\bar{u})$, where we have also used that Q is weakly sequentially closed for the second limit. For the third limit, note that for an arbitrary $\tilde{g} \in H$, by using (2.2), we get

$$|(Q_n(u_n) - Q(\bar{u}), \tilde{g})_H| \leq |(Q_n(u_n) - Q(u_n), \tilde{g})_H| + |(Q(u_n) - Q(\bar{u}), \tilde{g})_H|$$
$$\leq \epsilon_n \|g\|_H + |(Q(u_n) - Q(\bar{u}), \tilde{g})_H| \to 0,$$

where $(\cdot, \cdot)_H$ denotes the inner product in H.

Using the (weakly) lower semicontinuity of the norms, we have for every $u \in \mathcal{C}_{ad}$ that

$$\frac{1}{2} \|Q(\bar{u}) - g\|_H^2 + \frac{\alpha}{2} \|\bar{u}\|_U^2 \leq \liminf_n \left(\frac{1}{2} \|Q_n(u_n) - g\|_H^2 + \frac{\alpha}{2} \|u_n\|_U^2 \right)$$
$$\leq \lim_n \frac{1}{2} \|Q_n(u) - g\|_H^2 + \frac{\alpha}{2} \|u\|_U^2 = \frac{1}{2} \|Q(u) - g\|_H^2 + \frac{\alpha}{2} \|u\|_U^2.$$

Thus, we conclude that $\bar{u}$ is a minimizer of (1.2). We still need to show that $u_n \to \bar{u}$ strongly in U. Suppose there exists a $\mu > 0$ such that $\mu = \limsup_n \|u_n\|_U > \|\bar{u}\|_U$. Let $(u_{n_k})_{k \in \mathbb{N}}$ be a subsequence with $\|u_{n_k}\|_U \to \mu$ as $k \to \infty$. Then we have

$$\limsup_k \frac{1}{2} \|Q_{n_k}(u_{n_k}) - g\|_H^2 = \limsup_k \left(\frac{1}{2} \|Q_{n_k}(u_{n_k}) - g\|_H^2 + \frac{\alpha}{2} (\|u_{n_k}\|_U^2 - \mu^2) \right)$$
$$\leq \lim_k \frac{1}{2} \|Q_{n_k}(\bar{u}) - g\|_H^2 + \frac{\alpha}{2} (\|\bar{u}\|_U^2 - \mu^2) \tag{2.4}$$
$$= \frac{1}{2} \|Q(\bar{u}) - g\|_H^2 + \frac{\alpha}{2} (\|\bar{u}\|_U^2 - \mu^2) < \frac{1}{2} \|Q(\bar{u}) - g\|_H^2.$$

This contradicts the lower semicontinuity of the norm and $Q_n(u_n) \rightharpoonup Q(\bar{u})$. Thus, $\|u_n\|_U \to \|\bar{u}\|_U$ as $n \to \infty$. Together with the weak convergence $u_n \rightharpoonup \bar{u}$ we get $u_n \to \bar{u}$ strongly in U and further

$$\limsup_n \|Q_n(u_n) - g\|_H \leq \|Q(\bar{u}) - g\|_H \leq \liminf_n \|Q_n(u_n) - g\|_H.$$

Hence, $\lim_n \|Q_n(u_n)\|_H = \|Q(\bar{u})\|_H$, which implies the second limit in (2.3). $\square$

4

For a quantitative convergence result, we invoke the following assumptions which are motivated by the analysis of nonlinear inverse problems [23, 31].

Assumption 2.4. *Assume that Q is Fréchet differentiable and that there exists $L_0 > 0$ such that*

$$\|Q'(u)\|_{\mathcal{L}(U,H)} \leq L_0 \quad \text{for all } u \in \mathcal{C}_{ad}. \tag{2.5}$$

Assume further that the Fréchet derivative is locally Lipschitz with modulus $L_1 > 0$, i.e.,

$$\|Q'(u_a) - Q'(u_b)\|_{\mathcal{L}(U,H)} \leq L_1 \|u_a - u_b\|_U, \quad \text{for all } u_a, u_b \in \mathcal{C}_{ad}. \tag{2.6}$$

Moreover, let the Fréchet derivatives of Q and Q_n satisfy the following error bounds

$$\|Q'(u) - Q_n'(u)\|_{\mathcal{L}(U,H)} \leq \eta_n, \quad \text{for all } u \in \mathcal{C}_{ad}, \tag{2.7}$$

where $\eta_n \in (0,1)$ for all $n \in \mathbb{N}$ and $\eta_n \downarrow 0$. Finally, let the two constants L_0 and L_1 satisfy

$$L_0(L_0 + 1) + L_1 \|Q(\bar{u}) - g\|_H < \alpha, \tag{2.8}$$

with $\bar{u}$ being the minimizer of (1.2).

The condition in (2.5) indicates that

$$\|Q(u_a) - Q(u_b)\|_H \leq L_0 \|u_a - u_b\|_U, \quad \text{for all } u_a, u_b \in \mathcal{C}_{ad}. \tag{2.9}$$

Theorem 2.5. *Let the assumptions of Theorem 2.3 as well as Assumption 2.4 hold. Then, we have*

$$\|u_n - \bar{u}\|_U \leq \frac{1}{\alpha - L_0(L_0 + \eta_n) - L_1 \|Q(\bar{u}) - g\|_H} \left(L_0 \epsilon_n + \epsilon_n \eta_n + \|Q(\bar{u}) - g\|_H \eta_n \right). \tag{2.10}$$

Proof. First-order optimality yields

$$\bar{u} = \mathcal{P}_{\mathcal{C}_{ad}} \left(-(Q'(\bar{u}))^* w \right) \quad \text{and} \quad u_n = \mathcal{P}_{\mathcal{C}_{ad}} \left(-(Q_n'(u_n))^* w_n \right), \tag{2.11}$$

where $w = \frac{Q(\bar{u}) - g}{\alpha}$ and $w_n = \frac{Q_n(u_n) - g}{\alpha}$. The inequalities in (2.5), (2.6), (2.7), and (2.9) and the fact that $\|Q'(u)\|_{\mathcal{L}(U,H)} = \|(Q'(u))^*\|_{\mathcal{L}(H^*,U^*)}$ imply

$$
\begin{aligned}
\|u_n - \bar{u}\|_U &\leq \|(Q_n'(u_n))^* w_n - (Q'(\bar{u}))^* w\|_{U^*} \\
&\leq \|(Q_n'(u_n))^* (w_n - w)\|_{U^*} + \|((Q_n'(u_n))^* - (Q'(\bar{u}))^*) w\|_{U^*} \\
&\leq (L_0 + \eta_n) \|w_n - w\|_H + \|w\|_H \eta_n + L_1 \|w\|_H \|u_n - \bar{u}\|_U \\
&\leq \frac{L_0 + \eta_n}{\alpha} \|Q(\bar{u}) - Q_n(u_n)\|_H + \|w\|_H \eta_n + L_1 \|w\|_H \|u_n - \bar{u}\|_U \\
&\leq \frac{L_0 + \eta_n}{\alpha} (\epsilon_n + L_0 \|u_n - \bar{u}\|_U) + \|w\|_H \eta_n + L_1 \|w\|_H \|u_n - \bar{u}\|_U.
\end{aligned}
$$

Moving all terms that involve $\|u_n - \bar{u}\|_U$ to the left-hand side we get

$$\left(1 - \frac{L_0(L_0 + \eta_n)}{\alpha} - L_1 \|w\|_H\right) \|u_n - \bar{u}\|_U \leq \frac{L_0}{\alpha} \epsilon_n + \frac{\epsilon_n \eta_n}{\alpha} + \|w\|_H \eta_n.$$

Finally, using $w = \frac{Q(\bar{u}) - g}{\alpha}$ we find (2.10). $\square$

Observe that for $Q(\bar{u}) = g$ (perfect matching) the a priori bound is essentially controlled by ϵ_n only:

$$\|u_n - \bar{u}\|_U \leq \frac{L_0 + \eta_n}{\alpha - L_0(L_0 + \eta_n)} \epsilon_n.$$

Note further that the error bound depends on a sufficiently large α such that (2.8) is satisfied.

In the special case where $\mathcal{C}_{ad}$ is redundant, i.e., when $\mathcal{J}'(\bar{u}) = 0$, improved error bounds can be derived. This is in particular true for perfect matching which also allows to relax the conditions on α.

Theorem 2.6. *Let the assumptions of Theorem 2.3 hold and suppose that the Lipschitz condition (2.6) is satisfied with the constant L_1 such that*

$$L_1 \|Q(\bar{u}) - g\|_H < \alpha. \tag{2.12}$$

If $\mathcal{J}'(\bar{u}) = 0$, then for sufficiently large $n \in \mathbb{N}$ we have the following error bound

$$\|u_n - \bar{u}\|_U \leq \sqrt{\frac{3}{\alpha - L_1 \|Q(\bar{u} - g)\|_H}} \sqrt{\epsilon_n^2 + 2 \|Q(\bar{u}) - g\|_H^2}. \tag{2.13}$$

Proof. Since u_n is a minimizer for every $n \in \mathbb{N}$, we have that $\mathcal{J}_n(u_n) \leq \mathcal{J}_n(\bar{u})$ with $\mathcal{J}_n(u) := J(Q_n(u), u)$. Adding $\frac{\alpha}{2}(\|u_n - \bar{u}\|_U^2 - \|u_n\|_U^2)$ to both sides of the inequality gives

$$\frac{1}{2} \|Q_n(u_n) - g\|_H^2 + \frac{\alpha}{2} \|u_n - \bar{u}\|_U^2 \leq \frac{1}{2} \|Q_n(\bar{u}) - g\|_H^2 + \alpha \langle \iota_U^{-1} \bar{u}, \bar{u} - u_n \rangle_{U^*, U}. \tag{2.14}$$

Using Theorem 2.3, Taylor's expansion and (2.6), we get for sufficiently large $n \in \mathbb{N}$

$$Q(u_n) - Q(\bar{u}) = Q'(\bar{u})(u_n - \bar{u}) + q(u_n, \bar{u}), \text{ where } \|q(u_n, \bar{u})\|_H \leq \frac{L_1}{2} \|u_n - \bar{u}\|_U^2.$$

By our assumptions and first-order optimality we have $\bar{u} = -\iota_U(Q'(\bar{u}))^* w$ where $w = \alpha^{-1}(Q(\bar{u}) - g)$ with $L_1 \|w\|_H < 1$ because of (2.12). This leads to

$$
\begin{aligned}
\langle \iota_U^{-1} \bar{u}, \bar{u} - u_n \rangle_{U^*, U} &= (-w, Q'(\bar{u})(\bar{u} - u_n))_H \leq \|w\|_H \|Q'(\bar{u})(\bar{u} - u_n)\|_H \\
&\leq \|w\|_H \left(\frac{L_1}{2} \|u_n - \bar{u}\|_U^2 + \|Q(u_n) - Q_n(u_n)\|_H + \|Q_n(u_n) - g\|_H + \|g - Q(\bar{u})\|_H \right) \\
&\leq \frac{\|w\|_H L_1}{2} \|u_n - \bar{u}\|_U^2 + \frac{1}{2} \left(\alpha \|w\|_H^2 + \frac{1}{\alpha} \|Q_n(u_n) - g\|_H^2 \right) \\
&\quad + \left(\alpha \|w\|_H^2 + \frac{1}{2\alpha} \|Q(u_n) - Q_n(u_n)\|_H^2 + \frac{1}{2\alpha} \|g - Q(\bar{u})\|_H^2 \right),
\end{aligned}
\tag{2.15}
$$

where we have used the identity $ab \leq \frac{1}{2\alpha}a^2 + \frac{\alpha}{2}b^2$. Returning to (2.14) and using (2.15), we derive

$$
\begin{aligned}
\|u_n - \bar{u}\|_U^2 &\leq \frac{1}{\alpha} \|Q_n(\bar{u}) - g\|_H^2 + \|w\|_H L_1 \|u_n - \bar{u}\|_U^2 + 3\alpha \|w\|_H^2 \\
&\quad + \frac{1}{\alpha}(\|Q(u_n) - Q_n(u_n)\|_H^2 + \|g - Q(\bar{u})\|_H^2) \\
&\leq \frac{2}{\alpha} \|Q_n(\bar{u}) - Q(\bar{u})\|_H^2 + \|w\|_H L_1 \|u_n - \bar{u}\|_U^2 + 3\alpha \|w\|_H^2 \\
&\quad + \frac{1}{\alpha} \|Q(u_n) - Q_n(u_n)\|_H^2 + \frac{3}{\alpha} \|g - Q(\bar{u})\|_H^2.
\end{aligned}
\tag{2.16}
$$

Taking into account (2.12), we get

$$\|u_n - \bar{u}\|_U^2 \leq \frac{1}{(1 - \|w\|_H L_1)} \frac{3}{\alpha} \left(\epsilon_n^2 + \alpha^2 \|w\|_H^2 + \|g - Q(\bar{u})\|_H^2 \right),$$

for sufficiently large $n \in \mathbb{N}$. Replacing now $\|w\|_H$ by $\frac{\|g - Q(\bar{u})\|_H}{\alpha}$ yields (2.13). $\qquad \square$

Note that in the case of perfect matching $Q(\bar{u}) = g$, (2.13) becomes

$$\|u_n - \bar{u}\|_U \leq \epsilon_n \sqrt{\frac{3}{\alpha}} \quad \text{for sufficiently large } n \in \mathbb{N}. \tag{2.17}$$

As stated earlier, our aim is to use approximations $Q_n = Q_{\mathcal{N}_n} = A\Pi_{\mathcal{N}_n}$ resulting from artificial neural networks to replace the partially unknown exact control-to-state map Π and $Q = A\Pi$. Therefore, we next collect some fundamental properties of such neural network based approximations.

3 A brief primer on artificial neural networks (ANNs)

Here, we briefly review some (well-known) results for ANNs as they will be useful in what follows. For more introduction on ANNs, one may refer to many textbooks of this topic, e.g., [20]. We recall that a standard feedforward ANN with one hidden layer is a function $\mathcal{N} : \mathbb{R}^r \to \mathbb{R}^s$ of the following structure:

$$\mathcal{N}(x) = W_0 \sigma(W_1 x + b_1) + b_0, \quad x \in \mathbb{R}^r, \tag{3.1}$$

where $W_1 \in \mathbb{R}^{l \times r}$, $b_1 \in \mathbb{R}^l$, $W_0 \in \mathbb{R}^{s \times l}$ and $b_0 \in \mathbb{R}^s$. In that case we say that the hidden layer has l *neurons*. Here, $\sigma : \mathbb{R} \to \mathbb{R}$ is an infinitely differentiable activation function which acts component-wise on a vector in $\mathbb{R}^l$. In the output layer, the activation function is usually the identity map, therefore ignored in (3.1), while in the other hidden layers, it involves nonlinear transformations. Some standard smooth activation functions are the following ones:

- Sigmoid: a term denoting a family of functions, e.g., tansig ($\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$), logsig ($\sigma(z) = \frac{1}{1 + e^{-z}}$), arctan ($\sigma(z) = \arctan(z)$), etc.

- Probability functions, e.g., softmax ($\sigma_i(z) = \frac{e^{-z_i}}{\sum_j e^{-z_j}}$). Here the index i denotes the i-th neuron in a given layer, with the summation indexed by j being taken over all the neurons of the same layer.

We see that for the softmax function, neurons of the same layer may have different activition functions. Notice that the smoothness of the activation function is the one that determines the smoothness of $\mathcal{N}$.

Next we state a classical result, see, for instance, [37, Theorem 3.1]. Below "·" denotes the standard inner product in the underlying Euclidean space.

Theorem 3.1. *Let $\sigma \in C(\mathbb{R})$. Consider the following set*

$$R_\sigma := \left\{ \mathcal{N} : \mathbb{R}^r \to \mathbb{R} \,|\, \mathcal{N}(x) = w_0 \cdot \sigma(W_1 x + b_1), \text{ with } w_0 \in \mathbb{R}^l, \, W_1 \in \mathbb{R}^{l \times r}, \, b_1 \in \mathbb{R}^l, \, l \in \mathbb{N}^+ \right\}.$$

Then R_σ is dense in $C(\mathbb{R}^r)$ in the topology of uniform convergence on compact sets if and only if σ is not a polynomial function.

Hence, for any $\epsilon > 0$, and for any given function $f \in C(K)$, $K \subset \mathbb{R}^r$ compact, there exists a function $\mathcal{N} = \mathcal{N}^\epsilon \in R_\sigma$ such that

$$\max_{x \in K} |f(x) - \mathcal{N}^\epsilon(x)| < \epsilon.$$

This approximation property can be also carried over to the derivatives of a given smooth function; see, e.g., [37, Theorem 4.1].

Theorem 3.2. *Let $m = \max \left\{ |m^i| : i = 1, 2, \ldots, s \right\}$, where each m^i is a standard differentiation multi-index, and define $C^{m^1, \ldots, m^s}(\mathbb{R}^r) := \bigcap_{i=1}^s C^{m^i}(\mathbb{R}^r)$. Then R_σ is dense in $C^{m^1, \ldots, m^s}(\mathbb{R}^r)$ if $\sigma \in C^m(\mathbb{R})$ is not a polynomial function.*

As a consequence, for any $f \in C^{m^1, \ldots, m^s}(K)$, for every compact $K \subset \mathbb{R}^r$ and every $\epsilon > 0$, there exists a function $\mathcal{N} = \mathcal{N}^\epsilon \in R_\sigma$ such that

$$\max_{x \in K} \left| D^k f(x) - D^k \mathcal{N}^\epsilon(x) \right| < \epsilon,$$

for all multi-indices k such that $0 \le k \le m^i$ for some i.

Note that these results imply analogous error bounds for (3.1), i.e., for the vector-valued case. They can be also generalized to mutiple-hidden-layer networks as the next theorem shows, see [28].

Theorem 3.3. *A standard multi-layer feedforward network with a continuous activation function can uniformly approximate any continuous function to any degree of accuracy if and only if its activation function is not a polynomial.*

One of the main tasks of deep learning, a specific branch of machine learning, is to identify suitable choices for $W_0 \in \mathbb{R}^{s \times l_\ell}$, $W_1 \in \mathbb{R}^{l_1 \times r}$, $W_i \in \mathbb{R}^{l_i \times l_{i-1}}$ for $i = 2, \ldots, \ell$, and $b_0 \in \mathbb{R}^s$, $b_i \in \mathbb{R}^{l_i}$, where $i = 1, \ldots, \ell$ represents the i-th hidden layer of the underlying ANN, from a given data set $D = \{(x_j, f_j) \in$

$\mathbb{R}^r \times \mathbb{R}^s : j = 1, \ldots, n_D\}$, with $n_D \in \mathbb{N}$ sufficiently large. A typical approach in this context seeks to find a (global) solution to the nonconvex minimization problem

$$\text{minimize} \sum_{j=1}^{n_D} \mathfrak{d}(\mathcal{N}(x_j), f_j) + \mathfrak{r}(W, b) \quad \text{over } (W, b) \in \mathcal{F}_{ad}, \tag{3.2}$$

where $\mathcal{N}$ results from a multi-layer ANN that depends on $\Theta := (W, b)$, with $W := (W_0, \ldots, W_\ell)$ and $b := (b_0, b_1, \ldots, b_\ell)$. Further, $\mathfrak{d}$ denotes a suitable distance measure, $\mathfrak{r}$ is an optional regularization term inducing some a priori properties of Θ, and $\mathcal{F}_{ad}$ encodes possible additional constraints. While the study of (3.2) is an interesting and challenging subject in its own right, here we rather assume that the learning process, i.e., the computation of a suitable Θ, has been completed. We then study analytical properties of the resulting $\mathcal{N}$, or the solution map $\Pi_\mathcal{N}$ or $Q_\mathcal{N}$ in view of (1.2), in the context of our target applications and report on associated numerical results.

4 Application: Distributed control of semilinear elliptic PDEs

In our first application we consider the following model problem associated with the distributed optimal control of a semilinear elliptic PDE:

$$\text{minimize} \quad J(y, u) := \frac{1}{2}\|y - g\|_{L^2(\Omega)}^2 + \frac{\alpha}{2}\|u\|_{L^2(\Omega)}^2, \quad \text{over } (y, u) \in H^1(\Omega) \times L^2(\Omega) \tag{4.1}$$

$$\text{s.t.} \quad -\Delta y + f(x, y) = u \quad \text{in } \Omega, \quad \partial_\nu y = 0 \text{ on } \partial\Omega, \tag{4.2}$$

$$u \in \mathcal{C}_{ad} := \{v \in L^2(\Omega) : \underline{u}(x) \le v(x) \le \overline{u}(x), \quad \text{for a.e. } x \in \Omega\}, \tag{4.3}$$

where $\underline{u}, \overline{u}$ with $\underline{u} \le \overline{u}$ belong to $L^\infty(\Omega)$, and 'a.e.' stands for 'almost every' in the sense of the Lebesgue measure. Moreover, we have $g \in L^2(\Omega)$, and $\Omega \subset \mathbb{R}^d$, $d \ge 2$, is a bounded domain with Lipschitz boundary. In view of our general model problem class (1.1) we have $H = U = L^2(\Omega)$, $Y = H^1(\Omega)$, $Z = H^{-1}(\Omega)$, $A = \text{id}$, and e is given by the PDE in (4.2). For more details on the involved Lebesgue and Sobolev spaces we refer to [2]. Concerning f we invoke the following assumption throughout this section:

Assumption 4.1. *The nonlinear function $f = f(x, z) : \Omega \times \mathbb{R} \to \mathbb{R}$ is measurable with respect to x for every $z \in \mathbb{R}$ and continuously differentiable with respect to z for almost every $x \in \Omega$. There exists a function $F : \Omega \times \mathbb{R} \to \mathbb{R}$ so that $\partial_z F(\cdot, z) = f(\cdot, z)$. F and f satisfying the following conditions, for all $z \in \mathbb{R}$*

$$|f(\cdot, z)| \le b_1 + c_1 |z|^{p-1} \quad \text{and} \quad -f(\cdot, z)z + F(\cdot, z) \le b_2, \tag{4.4}$$

which combined also result to

$$F(\cdot, z) \le b_0 + c_0 |z|^p, \tag{4.5}$$

for some constants $b_0, b_1, b_2 \in \mathbb{R}$ and $c_0, c_1 > 0$ and for p with $1 < p \le \frac{2d}{d-2}$ for $d \ge 3$, $1 < p < +\infty$ for $d = 2$, or $1 < p \le +\infty$ for $d = 1$. The interpretation of $p = \infty$ for $d = 1$ is that the growth conditions in (4.4) are not required to hold. Finally, we assume that F is coercive in the sense that $\lim_{\|y\|_{L^p(\Omega)} \to \infty} \frac{\int_\Omega F(x,y)dx}{\|y\|_{L^p(\Omega)}} = \infty$, and F is bounded from below, i.e., $F(x, z) \ge F_0$ for some $F_0 \in \mathbb{R}$, for all $z \in \mathbb{R}$ and for almost every $x \in \Omega$.

The above assumption particular indicates that both f and F satisfy the Carathéodory condition, and thus induce some operators of Nemytskii type.

Moreover, observe also that the conditions on p enable the embedding $H^1(\Omega) \subset L^p(\Omega)$. Also note that the Assumption 4.1 is satisfied for $F(x, z) = \alpha(x)\pi_p(z)$ with $\alpha \in L^\infty(\Omega)$ and $\alpha(x) > 0$ for almost every $x \in \Omega$ and π_p being a polynomial of degree p and positive coefficient on the term of degree p; the latter being equal to $|z|^p$ if p is odd such that the coercivity assumption is not violated.

Given the above assumption, the PDE (4.2) is related to the variational problem

$$\text{minimize} \quad G(y) := \frac{1}{2}\|\nabla y\|_{L^2(\Omega)}^2 + \int_\Omega F(x, y)\, dx - \int_\Omega uy\, dx \quad \text{over } y \in H^1(\Omega). \tag{4.6}$$

A particular example is given by a Ginzburg-Landau model for superconductivity where $f(z) = \eta^{-1}(z^3 - z)$ with a parameter $\eta > 0$. It gives rise to the double-well type variational model

$$\text{minimize} \quad \frac{1}{2}\|\nabla y\|_{L^2(\Omega)}^2 + \frac{1}{4\eta}\int_\Omega (y^2 - 1)^2 dx - \int_\Omega uy\, dx \quad \text{over } y \in H^1(\Omega), \tag{4.7}$$

for given $u \in L^2(\Omega)$ or in fact, to a more a general space. The next proposition shows existence of solutions for (4.6).

Proposition 4.2. *Let Assumption 4.1 hold, and suppose that $u \in L^r(\Omega)$ for some $r \geq \frac{p}{p-1}$. Then the optimization problem (4.6) admits a solution in $H^1(\Omega)$.*

Proof. Notice that due to the coercivity assumption we can find a $C > 0$ such that $\|u\|_{L^r(\Omega)} < CC_1$ with C_1 being the constant involved in the embedding $L^p(\Omega) \subset L^{\frac{r}{r-1}}(\Omega)$ such that

$$\int_\Omega F(x,y)\, dx - \int_\Omega uy\, dx \geq C\|y\|_{L^p(\Omega)} - \|u\|_{L^r(\Omega)}\|y\|_{L^{\frac{r}{r-1}}(\Omega)}$$
$$\geq (CC_1 - \|u\|_{L^r(\Omega)})\|y\|_{L^{\frac{r}{r-1}}(\Omega)} \geq 0, \tag{4.8}$$

provided $\|y\|_{L^p(\Omega)}$ is large enough. This together with the lower bound $F \geq F_0$ implies that the energy G is bounded from below and thus there is an infimizing sequence $(y_n)_{n\in\mathbb{N}} \in H^1(\Omega) \subset L^p(\Omega)$. Using the above inequality one easily deduces that $\|y_n\|_{L^{\frac{r}{r-1}}(\Omega)}$ is bounded, and with the help of the Poincaré inequality a uniform $H^1(\Omega)$ bound is also obtained for that sequence. Therefore, we only need to show that $G(\cdot)$ is weakly lower semicontinuous in $H^1(\Omega)$. For this, it suffices to check the term involving F, since the arguments for the other two terms are straightforward. Assuming $y_n \rightharpoonup y$ in $H^1(\Omega)$, by the compact embedding of $H^1(\Omega) \hookrightarrow L^1(\Omega)$, we have that $y_n \to y$ almost everywhere, up to a subsequence. Due to the continuity of F with respect to the second variable, we have $F(\cdot, y) = \lim_{n\to\infty} F(\cdot, y_n)$ almost everywhere. Since $F(\cdot, y_n), F(\cdot, y) \geq F_0$, by Fatou's lemma we have

$$\int_\Omega F(x,y)\, dx \leq \liminf_{n\to\infty} \int_\Omega F(x, y_n)\, dx,$$

and thus $G(\cdot)$ is weakly lower semicontinuous. $\qquad\square$

Before we proceed, it is useful to recall the following standard result on linear elliptic PDEs [18, 44].

Theorem 4.3. *Let $v \in L^r(\Omega)$, $a \in L^\infty(\Omega)$ with $a > 0$. Then the following equation admits a unique solution*

$$-\Delta s + as = v \quad \text{in } \Omega, \qquad \partial_\nu s = 0 \text{ on } \partial\Omega.$$

Furthermore there exist constants $C_h > 0$ and $C_l > 0$ independent of a and v such that

$$\|s\|_{H^1(\Omega)} \leq C_h \|v\|_{L^r(\Omega)} \quad \text{and} \quad \|s\|_{C(\overline{\Omega})} \leq C_l \|v\|_{L^r(\Omega)}. \tag{4.9}$$

Using the polynomial growth of F together with the continuous embedding $H^1(\Omega) \subset L^{\frac{r}{r-1}}(\Omega)$, one verifies the Fréchet differentiability of $G : H^1(\Omega) \to \mathbb{R}$. The Euler-Lagrange equation associated with (4.6) is given by

$$-\Delta y + f(x,y) = u \quad \text{in } \Omega, \qquad \partial_\nu y = 0 \text{ on } \partial\Omega, \tag{4.10}$$

and it is satisfied for every solution y of (4.6). Under Assumption 4.1, the solutions of (4.10) can be uniformly bounded with respect to $\|\cdot\|_{C(\overline{\Omega})}$, as shown next.

Proposition 4.4. *Let the Assumption 4.1 be satisfied, and let $\mathcal{C}_{ad} \subset L^\infty(\Omega)$ be bounded. Then there exists a constant $K > 0$ such that for all solutions of (4.10), it holds*

$$\|y\|_{H^1(\Omega)} + \|y\|_{C(\overline{\Omega})} \leq K, \quad \text{for all } u \in \mathcal{C}_{ad}. \tag{4.11}$$

Proof. From the fact that $y \in L^p(\Omega)$, the growth condition (4.4) and the measurability of f, we have $f(\cdot, y) \in L^{\frac{p}{p-1}}(\Omega)$. We can rewrite (4.10) in the following form

$$-\Delta y + \epsilon y = u + \epsilon y - f(x,y) \quad \text{in } \Omega, \qquad \partial_\nu y = 0 \text{ on } \partial\Omega, \tag{4.12}$$

9

for some $\epsilon > 0$. Let us define $\tilde{r} := \min\left\{\frac{r}{r-1}, \frac{p}{p-1}\right\}$. Then $u + \epsilon y + f(\cdot, y) \in L^{\tilde{r}}(\Omega)$ since $u \in \mathcal{C}_{ad} \subset L^\infty(\Omega)$. Applying (4.9) to (4.12) yields

$$\|y\|_{H^1(\Omega)} + \|y\|_{C(\overline{\Omega})} \le (C_h + C_l)\left(\|u\|_{L^{\tilde{r}}(\Omega)} + \epsilon\|y\|_{L^{\tilde{r}}(\Omega)} + \|f(\cdot, y)\|_{L^{\tilde{r}}(\Omega)}\right). \tag{4.13}$$

As all solutions of (4.12) are stationary points of G, in view of (4.4), every weak solution y satisfies

$$G(y) = \frac{1}{2}\|\nabla y\|^2_{L^2(\Omega)} + \int_\Omega F(x, y)\, dx - \int_\Omega uy = \int_\Omega -f(x, y)y + F(x, y)\, dx \le b_2|\Omega|, \tag{4.14}$$

where we use the weak formulation of (4.12) tested with y.

Using the coercivity of G, we can find some constant $M > 0$ independent of y such that $\|y\|_{L^p(\Omega)} \le M$. Since $(p-1)\tilde{r} \le p$, by (4.4), we have

$$\|f(\cdot, y)\|_{L^{\tilde{r}}(\Omega)} \le d_0 + d\left\|y^{p-1}\right\|_{L^{\tilde{r}}(\Omega)} \le d_0 + \tilde{d}\|y\|^{p-1}_{L^p(\Omega)} \le \tilde{M}. \tag{4.15}$$

Returning to (4.13), we choose a sufficiently small $\epsilon > 0$ such that the second term on the right-hand side of (4.13) is absorbed by $\|y\|_{H^1(\Omega)}$. Since $L^{\tilde{r}}(\Omega) \subset L^\infty(\Omega)$ and $\mathcal{C}_{ad}$ is bounded, $\|u\|_{L^{\tilde{r}}(\Omega)}$ is uniformly bounded for all $u \in \mathcal{C}_{ad}$. Finally, taking into account (4.13) and (4.15) we have

$$\|y\|_{H^1(\Omega)} + \|y\|_{C(\overline{\Omega})} \le (\tilde{C}_h + \tilde{C}_l)(\|u\|_{L^{\tilde{r}}(\Omega)} + \tilde{M}) \le K, \tag{4.16}$$

which is the conclusion. $\qquad\square$

Notice that for monotone f, one can directly refer to standard results in the literature, e.g., [44], where uniform bounds on the solution of (4.10) are shown for that case.

4.1 Continuity and sensitivity of the control-to-state map

Since $f(\cdot, \cdot)$ might be nonmonotone with respect to the second variable, this may give rise to a lack of uniqueness of a solution to the semilinear PDE (4.2). In the monotone case, the continuity result is more direct to show, thus we focus on the nonmonotone case here.

Under our standing assumptions, (4.2) has a nonempty set of solutions y satisfying

$$\|y\|_{H^1(\Omega)} + \|y\|_{C(\overline{\Omega})} \le K$$

for some constant K independent of u since $\mathcal{C}_{ad}$ is bounded. The associated continuity result stated next, relies on a Γ–convergence technique. We note that for this section we take $r = 2$.

Proposition 4.5. *Let $u_n \to u$ in $L^2(\Omega)$ and $G_n, G : H^1(\Omega) \to \mathbb{R}$ be the corresponding energies in (4.6). Then G_n Γ–converges to G with respect to the H^1 topology. Furthermore, G_n is equi-coercive.*

Proof. Observe first that one easily checks that G_n Γ–converges to G. This is because the function $\frac{1}{2}\|\nabla(\cdot)\|^2_{L^2(\Omega)} + \int_\Omega F(x, \cdot)\, dx$ is weakly lower semicontinuous with respect to the $H^1(\Omega)$ convergence (and hence it Γ–converges to itself), while the function $y \mapsto \int_\Omega u_n y\, dx$ continuously converges to the function $y \mapsto \int_\Omega uy\, dx$ (see [13, Def. 4.7] for the notion of continuous convergence). The assertion follows from the stability of Γ-convergence under continuous perturbations [13, Prop. 6.20].

In order to see that G_n is equi-coercive, it suffices to find a lower semicontinuous coercive function $\Psi : H^1(\Omega) \to \mathbb{R}$ such that $G_n \ge \Psi$ on $H^1(\Omega)$, cf. [13, Prop. 7.7]. This follows from the fact that $(\|u_n\|_{L^2(\Omega)})_{n \in \mathbb{N}}$ is a bounded sequence and from the coercivity condition in Assumption (4.1), see also (4.8). $\qquad\square$

With the help of Γ–convergence and equi-coercivity one can get the classical results on Γ–convergence with respect to global and local minimizers. It is of particular interest whether y_0 is an isolated local minimizer of G (and in particular satisfies (4.2)). In this case there exists a sequence $\tilde{y}_n$ with $\tilde{y}_n \to y_0$ in $H^1(\Omega)$ such that for all sufficiently large n, $\tilde{y}_n$ is a local minimizer of G_n (hence it also satisfies (4.2)); see [11]. This implies that if $u_n \to u_0$ in $L^2(\Omega)$ and $y_0 \in \Pi(u_0)$ is an isolated local minimizer of G, then there exists a sequence $(y_n)_{n \in \mathbb{N}}$ in $H^1(\Omega)$ such that $y_n \in \Pi(u_n)$ and $y_n \to y_0$ in $H^1(\Omega)$.

Remark 4.6. *We note that solutions of the PDE (4.2) are not necessarily local minimizers of the variational problem (4.6). In order to make sure that y_0 is an isolated local minimizer, one can check second-order conditions on (4.6). In this context, second-order sufficiency relates to $(s, -\Delta s + \partial_y f(\cdot, y_0)s) > \epsilon \|s\|_{H^1(\Omega)}^2$ for all $s \in H^1(\Omega)$ with some $\epsilon > 0$. Therefore, if $f(\cdot, \cdot)$ is a strictly monotone function with respect to its second variable, then the positive definiteness condition is automatically guaranteed. For the more general case, it turns out that a similar, but yet milder condition (see (4.19) below) helps to establish the sensitivity result for the control-to-state map.*

Given this approximating sequence $(y_n)_{n\in\mathbb{N}}$ for $y_0 \in \Pi(u_0)$, convergence rates and differentiability of the control-to-state map in a certain sense are shown next. For this, we also assume that

$$\forall M > 0 \ \exists L_M > 0 : |\partial_y f(x, y_1) - \partial_y f(x, y_2)| \leq L_M |y_1 - y_2|, \tag{4.17}$$

for almost every $x \in \Omega$ and for all $y_1, y_2 \in [-M, M]$. This also implies

$$\forall M > 0 \ \exists C > 0 : \quad |\partial_y f(x, y)| < C \text{ for a.e. } x \in \Omega \text{ and } \forall y \in [-M, M]. \tag{4.18}$$

Theorem 4.7. *Assume that (4.17) holds for f, let $\Pi : L^2(\Omega) \rightrightarrows H^1(\Omega)$ be the possibly multi-valued control-to-state map of (4.2) and fix some $u_0, h \in L^2(\Omega) \cap \mathcal{C}_{ad}$ as well as $y_0 \in \Pi(u_0)$. Define $(\partial_y f(\cdot, y_0))^- := \min\{\partial_y f(\cdot, y_0), 0\}$, and assume that*

$$\left\|(\partial_y f(\cdot, y_0))^-\right\|_{L^2(\Omega)} < \frac{1}{C_l} \quad and \quad \left\|(\partial_y f(\cdot, y_0))^-\right\|_{L^\infty(\Omega)} < \frac{1}{C_h}, \tag{4.19}$$

where C_l and C_h are the positive constants defined in (4.9). Suppose $u_n = u_0 + t_n h$ for a sequence $t_n \to 0$, and suppose there exists $y_n \in \Pi(u_n)$ with $y_n \to y_0$ in $H^1(\Omega)$. Then we have

$$\|y_n - y_0\|_{H^1(\Omega)} \leq C t_n, \tag{4.20}$$

for some constant C and large enough $n \in \mathbb{N}$. Moreover, one has that every weak cluster point of $\frac{y_n - y_0}{t_n}$, denoted by p, solves the following linear PDE

$$-\Delta p + \partial_y f(\cdot, y_0)p = h \quad in \ \Omega, \qquad \partial_\nu p = 0 \ \ on \ \partial\Omega.$$

In particular, p satisfies the energy bounds:

$$\|p\|_{H^1(\Omega)} \leq C_H \|h\|_{L^2(\Omega)} \quad and \quad \|p\|_{C(\overline{\Omega})} \leq C_c \|h\|_{L^2(\Omega)}, \tag{4.21}$$

with constants C_H and C_c depending on C_h and C_l.

Proof. Subtracting the equations that correspond to the pairs (u_n, y_n) and (u_0, y_0) and using the mean value theorem, we get

$$-\Delta(y_n - y_0) = t_n h + f(\cdot, y_0) - f(\cdot, y_n) = t_n h - \partial_y f(\cdot, y_0 + \gamma_h(y_n - y_0))(y_n - y_0), \tag{4.22}$$

where $\gamma_h \in L^\infty(\Omega)$ with $\|\gamma_h\|_{L^\infty(\Omega)} \leq 1$, see Remark 4.10 regarding measurability of such γ_h. Note that $y_n, y_0 \in C(\overline{\Omega})$ with a uniform bound $K > 0$, therefore from (4.18) we have $\partial_y f(\cdot, y_0 + \gamma_h(y_n - y_0)) \in L^\infty(\Omega)$. Then, given $\epsilon > 0$, we rewrite (4.22) as

$$-\Delta(y_n - y_0) + (\epsilon + (\partial_y f(\cdot, \xi_n^h))^+)(y_n - y_0) = t_n h + (\epsilon + (\partial_y f(\cdot, \xi_n^h))^+ - \partial_y f(\cdot, \xi_n^h))(y_n - y_0), \tag{4.23}$$

where $\xi_n^h := y_0 + \gamma_h(y_n - y_0)$, and $(\partial_y f(\cdot, \xi_n^h))^+ = \max\{\partial_y f(\cdot, \xi_n^h), 0\}$. Now, using (4.9), we have

$$\begin{aligned}
&\frac{\epsilon}{C_h} \|y_n - y_0\|_{H^1(\Omega)} + \|y_n - y_0\|_{L^\infty(\Omega)} \\
&\leq (\epsilon + C_l)\left(t_n \|h\|_{L^2(\Omega)} + \left\|(\epsilon + (\partial_y f(\cdot, \xi_n^h))^+ - \partial_y f(\cdot, \xi_n^h))(y_n - y_0)\right\|_{L^2(\Omega)}\right) \\
&\leq (\epsilon + C_l)\left(t_n \|h\|_{L^2(\Omega)} + \left\|\epsilon + (\partial_y f(\cdot, \xi_n^h))^-\right\|_{L^2(\Omega)} \|y_n - y_0\|_{L^\infty(\Omega)}\right).
\end{aligned} \tag{4.24}$$

The last inequality holds since both y_n and y_0 are $C(\overline{\Omega})$ functions. Because $y_n \to y_0$ in $H^1(\Omega)$, we also have that $\xi_n^h \to y_0$ in $L^2(\Omega)$. From the continuity of $\partial_y f(x, \cdot)$, the fact that y_n, y_0 are uniformly

bounded in $C(\overline{\Omega})$ and from dominated convergence, we have that $\partial_y f(\cdot, \xi_n^h) \to \partial_y f(\cdot, y_0)$ in $L^2(\Omega)$. Thus, because of (4.19), there exists $\epsilon = \epsilon_0$ small enough, such that for sufficiently large n, we have $(\epsilon_0 + C_l) \left\| \epsilon_0 + (\partial_y f(\cdot, \xi_n^h))^- \right\|_{L^2(\Omega)} \le 1$. Then (4.24) leads to

$$\|y_n - y_0\|_{H^1(\Omega)} \le \frac{C_h(\epsilon_0 + C_l)}{\epsilon_0} \|h\|_{L^2(\Omega)} \, t_n. \tag{4.25}$$

From the above inequalities we have that $(\frac{y_n - y}{t_n})_{n \in \mathbb{N}}$ is uniformly bounded in $H^1(\Omega)$ and therefore admits a weakly convergent subsequence (unrelabelled) with weak limit p. Then, dividing by t_n and letting $t_n \to 0$ in (4.22), we have that p satisfies the following equation

$$-\Delta p + \partial_y f(\cdot, y_0) p = h \quad \text{in } \Omega, \quad \partial_\nu p = 0 \text{ on } \partial\Omega. \tag{4.26}$$

Note that (4.25) readily implies the first energy bound in (4.21). For the second bound in (4.21), the procedure is similar. For this we consider

$$\begin{aligned}
&\|y_n - y_0\|_{H^1(\Omega)} + \frac{\epsilon}{C_l} \|y_n - y_0\|_{C(\overline{\Omega})} \\
&\le (\epsilon + C_h) \left(t_n \|h\|_{L^2(\Omega)} + \left\| \epsilon + (\partial_y f(\cdot, \xi_n^h))^- \right\|_{L^\infty(\Omega)} \|y_n - y_0\|_{L^2(\Omega)} \right).
\end{aligned} \tag{4.27}$$

Invoking now the second condition in (4.19), and using exactly the same steps as for the first bound of (4.21), we find some $\epsilon_0' > 0$ to conclude the second bound in (4.21) when n is sufficiently large. $\qquad\square$

Remark 4.8. *The proof of Theorem 4.7 provides an alternative strategy for proving existence and energy estimates of solutions for certain type of linear elliptic PDEs, e.g. as in (4.26) when the elliptic coercivity is mildly violated. Also note that in the monotone case, $(\partial_y f(\cdot, y_0))^- \equiv 0$, and thus the conditions in (4.19) are always fulfilled.*

4.2 Existence results for learning-informed semilinear PDEs

As motivated in the introduction, in many applications the precise form of f is not known explicitly, but rather it can be inferred from given data only. Here we are particularly interested in neural networks to learn the hidden physical law or nonlinear mapping from such data. The corresponding existence result for PDEs that include such neural network approximations is stated next.

Proposition 4.9. *Let $f : \Omega \times \mathbb{R} \to \mathbb{R}$ and $F : \Omega \times \mathbb{R} \to \mathbb{R}$ be given as in Assumption 4.1 with the extra assumption that $f \in C(\overline{\Omega} \times \mathbb{R})$. Then, for every $\epsilon > 0$ there exists a neural network $\mathcal{N} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ such that*

$$\sup_{\|y\|_{L^\infty(\Omega)} < K} \|f(\cdot, y) - \mathcal{N}(\cdot, y)\|_U < \epsilon, \tag{4.28}$$

with K cf. (4.11). Moreover, the learning-informed PDE

$$-\Delta y + \mathcal{N}(\cdot, y) = u \quad \text{in } \Omega, \qquad \partial_\nu y = 0 \text{ on } \partial\Omega, \tag{4.29}$$

admits a weak solution which also satisfies (4.11) for sufficiently small $\epsilon > 0$.

Proof. From Theorem 3.1 we have that for every $\tilde{\epsilon} > 0$ there exists a neural network $\mathcal{N} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ such that $|f(x, y) - \mathcal{N}(x, y)| < \tilde{\epsilon}$ for every $(x, y) \in \overline{\Omega} \times [-K - 1, K + 1]$.

Thus, the existence of $\mathcal{N}$ such that (4.28) holds can be directly shown; note that $U = L^\infty(\Omega)$ is feasible in (4.28).

Consider next the function $N : \Omega \times \mathbb{R} \to \mathbb{R}$ given by

$$N(x, t) := \begin{cases} \int_0^t \mathcal{N}(x, s) \, ds + F(x, 0), & -(K + 1) \le t \le K + 1, \\ r_0(x) + F(x, t), & t > K + 1, \\ r_1(x) + F(x, t), & t < -(K + 1), \end{cases}$$

with $r_0(x) := \int_0^{K+1} \mathcal{N}(x, s) \, ds + F(x, 0) - F(x, K + 1)$, $r_1(x) := \int_0^{-K-1} \mathcal{N}(x, s) \, ds + F(x, 0) - F(x, -K - 1)$. Notice that $N(x, t)$ is continuous with $|N(x, t) - F(x, t)| < \epsilon(K + 1)$ for every $t \in \mathbb{R}$ and $x \in \Omega$. Next we apply some smoothing of $N(x, \cdot)$ in a small neighbourhood of $\Omega \times \{-K - 1\}$ and $\Omega \times \{K + 1\}$ such that

the previous approximation estimate still holds true, and continue to use the symbol N for the result. Then $N(x, \cdot)$ is differentiable with respect to the second variable for every $x \in \Omega$. Consider now the minimization problem

$$\inf_{y \in H^1(\Omega)} \frac{1}{2} \|\nabla y\|_{L^2(\Omega)}^2 + \int_\Omega N(x, y)\, dx - \int_\Omega u y\, dx. \tag{4.30}$$

One can now prove existence of a solution to (4.30) analogously to the proof of Proposition 4.2 for (4.6). We can show that the functional in $y \mapsto \int_\Omega N(x, y)\, dx$ is Frechét differentiable in $H^1(\Omega)$ with Frechét derivative $h \mapsto \int_\Omega \partial_y N(x, y) h\, dx$, see discussion after this proof. Thus, any solution to (4.30) satisfies the PDE

$$-\Delta y + \partial_y N(\cdot, y) = u, \quad \text{in } \Omega, \quad \partial_\nu y = 0 \text{ on } \partial\Omega. \tag{4.31}$$

By following estimates analogous to the ones leading to (4.11), we have in view of (4.15)–(4.16) and (4.28), that any solution y_0 also satisfies $\|y_0\|_{C(\overline{\Omega})} < K$ when ϵ is sufficiently small. Since $\partial_y N = \mathcal{N}$ on $\Omega \times [-K, K]$ we conclude that y_0 is a solution of (4.29). $\qquad\square$

Concerning the announced differentiability of $\Phi_N(y) := \int_\Omega N(x, y)\, dx$, define

$$\Phi_N'(y)h := \int_\Omega \partial_y N(x, y) h\, dx.$$

Since $\frac{|\Phi_N(y+h) - \Phi_N(y) - \Phi_N'(y)h|}{\|h\|_{H^1(\Omega)}} = \frac{|\Phi_N'(y + \tau_h h)h - \Phi_N'(y)h|}{\|h\|_{H^1(\Omega)}}$ for some $\tau_h \in L^\infty(\Omega)$ with $\|\tau_h\|_{L^\infty(\Omega)} \leq 1$, using the mean value theorem along with $H^1(\Omega) \subset L^{\frac{r}{r-1}}(\Omega)$, we have for a $C > 0$

$$\frac{|\Phi_N(y+h) - \Phi_N(y) - \Phi_N'(y)h|}{\|h\|_{H^1(\Omega)}} \leq C \left\| \partial_y(N(\cdot, y + \tau_h h) - N(\cdot, y)) \right\|_{L^r(\Omega)}. \tag{4.32}$$

Note that by definition, the growth rate of $N(x, \cdot)$ outside of $[-K - 1, K + 1]$ is exactly the same as the one of $F(x, \cdot)$. Therefore $\partial_y N(\cdot, y)$ is indeed an element of $L^r(\Omega)$. Finally, we need to verify that

$$\lim_{h \to 0} \|\partial_y N(y + \tau_h h) - \partial_y N(y)\|_{L^r(\Omega)} = 0 \quad \text{for } h \in H^1(\Omega).$$

This is true due to the continuity of the Nemytskii operator $\partial_y N : L^{\frac{r}{r-1}}(\Omega) \to L^r(\Omega)$.

Remark 4.10. *Notice that in (4.32) the mean value theorem is applied for every $x \in \Omega$ and τ_h is defined as a selector function of the multi-valued map $\tau : \Omega \rightrightarrows [0, 1]$ with*

$$\tau(x) = \{\lambda \in [0, 1] : N(x, y(x) + h(x)) - N(x, y(x)) - \partial_y N(x, y(x) + \lambda h(x))h(x) = 0\}.$$

Even though by definition τ_h is a bounded function, one still needs to show its measurability such that $\tau_h \in L^\infty(\Omega)$. Such a measurable selector function is indeed guaranteed by the Kuratowski–Ryll–Nardzewski selection theorem [4, Theorem 18.13] whose conditions can be verified in our case. In fact, we may choose $\tau_h(x) := \max \tau(x)$; see [4, Theorem 18.19]. We note that one can also use another version of mean value theorem where $\tau_h \in [0, 1]$ is just a real number. Since we use the former mean value theorem several times in the paper, for generality, we consider the point-wise version.

Note that the above set-up covers a wide range of problems, including the class of problems where the nonlinear function $f(\cdot, \cdot)$ is strictly monotone with respect to the second variable. In that case, the nonlinear PDE (4.2) admits a unique solution [44]. We also point out that in the monotone case direct methods allow to prove the existence of solutions and energy bounds for a wider array of monotone nonlinearities (such as, e.g., exponential functions). Moreover in that case, the regularity and growth conditions on the nonlinear function f can be relaxed. However, as pursuing such a generality is not the focus of the current paper, we skip detailed discussions here. We note however that structural aspects of the control problem such as first-order optimality, adjoints etc. remain intact even under relaxed conditions.

In order to give an example on this, we show in the next proposition how strict monotonicity for the learning-based model can indeed be preserved.

Proposition 4.11. *Let $f : \Omega \times \mathbb{R} \to \mathbb{R}$ satisfy Assumption 4.1 and $\partial_y f(x,y) \geq C_f$ for almost every $x \in \Omega$ and $y \in \mathbb{R}$ for some $C_f > 0$. We additionally assume that $f \in C(\Omega \times \mathbb{R})$. Then for every $\epsilon > 0$, for every compact set $\Omega_c \subset \Omega$, and for every $M > 0$, there exists a neural network $\mathcal{N} := \mathcal{N}^\epsilon_{\Omega_c,M} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ such that*

$$|f(x,z) - \mathcal{N}(x,z)| < \epsilon, \quad \text{for every } x \in \Omega_c \text{ and every } z \in [-M, M], \tag{4.33}$$

$$\partial_z \mathcal{N}(x,z) \geq C_\mathcal{N}, \quad \text{for all } x \in \Omega \text{ and } z \in [-M, M] \text{ for some } C_\mathcal{N} > 0. \tag{4.34}$$

If $f \in C^1(\Omega \times \mathbb{R})$, then we have in addition that

$$|\partial_z f(x,z) - \partial_z \mathcal{N}(x,z)| < \epsilon, \quad \text{for all } x \in \Omega_c \text{ and } z \in [-M, M]. \tag{4.35}$$

Proof. Let $\epsilon > 0$, $\Omega_c \subset \Omega$ compact, and $M > 0$. Further, let $\tilde{f} : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ be the extension by zero of f outside $\Omega \times \mathbb{R}$, ρ_δ a standard mollifier [6, Sec.2.2.2], and $\tilde{f}_\delta := \tilde{f} * \rho_\delta : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$. Next we choose $\delta > 0$ such that the following hold true: (i) $\bar{B}(x,\delta) := \{\hat{x} \in \mathbb{R}^d : \|\hat{x} - x\|_2 \leq \delta\} \subset \Omega$ for every $x \in \Omega_c$, (ii) $\tilde{f}_\delta(x,y) = f_\delta(x,y)$ for $(x,y) \in \Omega_c \times \mathbb{R}$, and (iii) $|f(x,y) - \tilde{f}_\delta(x,y)| < \epsilon/2$ for every $x \in \Omega_c$, $y \in [-M, M]$. Moreover, one finds that for sufficiently small $\delta > 0$ it holds that $\partial_y \tilde{f}_\delta(x,y) \geq C_{\tilde{f}}$ for some $C_{\tilde{f}} > 0$ for all $x \in \Omega$, $y \in \mathbb{R}$. Indeed, note that Assumption 4.1 and the mean value theorem yield for almost every $x' \in \Omega$, $y_1 < y_2$

$$f(x', y_2) - f(x', y_1) \geq C_f(y_2 - y_1). \tag{4.36}$$

Hence, using $\rho_\delta(\cdot) = \delta^{-(d+1)}\rho(\cdot/\delta)$ [6, Sec.2.2.2], we have

$$\tilde{f}_\delta(x, y_1) = \int_{B_\delta(x,y_1)\cap(\Omega\times\mathbb{R})} \tilde{f}(x',y')\delta^{-d-1}\rho\left(\frac{(x,y_1) - (x',y')}{\delta}\right) d(x',y')$$

$$\leq \int_{B_\delta(x,y_2)\cap(\Omega\times\mathbb{R})} \left(\tilde{f}(x',y') - C_f(y_2 - y_1)\right)\delta^{-d-1}\rho\left(\frac{(x,y_2) - (x',y')}{\delta}\right) d(x',y')$$

$$= \tilde{f}_\delta(x, y_2) - C_f\bigg(\underbrace{\int_{B_\delta(x,y_2)\cap(\Omega\times\mathbb{R})} \delta^{-d-1}\rho\left(\frac{(x,y_2) - (x',y')}{\delta}\right) d(x',y')}_{=:\tilde{C}}\bigg)(y_2 - y_1)$$

$$= \tilde{f}_\delta(x, y_2) - C_f\tilde{C}(y_2 - y_1).$$

We now use the fact that the boundary of Ω is Lipschitz to deduce that for some small enough $\delta > 0$ we have $\tilde{C} := \tilde{C}_{x,y} > c$ for some $c > 0$, for every $x \in \Omega$, $y \in \mathbb{R}$, and set $C_{\tilde{f}} := C_f c$. Hence from the last inequality above we deduce $\partial_y \tilde{f}_\delta(x,y) \geq C_{\tilde{f}}$. Utilizing now Theorems 3.1 and 3.2 for the compact set $\overline{\Omega} \times [-M, M] \subset \mathbb{R}^d \times \mathbb{R}$, we find a neural network $\mathcal{N} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ such that $|\tilde{f}_\delta(x,y) - \mathcal{N}(x,y)| < \epsilon/2$ as well as $|\partial_y \tilde{f}_\delta(x,y) - \partial_y \mathcal{N}(x,y)| < C_{\tilde{f}}/4$ for every $x \in \overline{\Omega}$ and $y \in [-M, M]$. Then with the use of the triangle inequality we get (4.33) and (4.34) for $C_\mathcal{N} = \frac{3}{4}C_{\tilde{f}}$.

Finally, when f is also continuously differentiable in $\Omega \times \mathbb{R}$, we can proceed as before with the extra care to choose $\delta > 0$ such that $|\partial_y f(x,y) - \partial_y \tilde{f}_\delta(x,y)| < \epsilon/2$ for every $x \in \Omega_c$, $y \in [-M, M]$. $\quad\square$

Note that if f is bounded on $\Omega \times [-K, K]$, for instance if $f \in C(\overline{\Omega} \times \mathbb{R})$ as in Proposition 4.9, then the estimate (4.28) holds here as well and if analogous conditions hold for the derivative of f then with the help of (4.35) we also have

$$\sup_{\|y\|_{L^\infty(\Omega)} < K} \|\partial_y f(\cdot, y) - \partial_y \mathcal{N}(\cdot, y)\|_U \leq \epsilon. \tag{4.37}$$

4.3 Error analysis for the control-to-state map

Our next target is to show the error bounds (2.2) and (2.7) for the solution maps (control-to-state maps) of the learning-informed versus the original PDE. Before we proceed, we first show the local Lipschitz conditions (2.9) and (2.6). For the ease of presentation we confine ourselves to a strict monotone $f(x, \cdot)$ here. For the nonmonotone $f(x, \cdot)$, we would require (4.19) to be satisfied for solutions uniformly bounded by K. Consider the following pairs of equations for $i \in \{1, 2\}$

$$\begin{cases} -\Delta y_i + f(\cdot, y_i) = u_i & \text{in } \Omega, \\ \quad \partial_\nu y_i = 0 & \text{on } \partial\Omega, \end{cases} \quad \text{and} \quad \begin{cases} -\Delta p_i + \partial_y f(\cdot, \bar{y}_i)p_i = v & \text{in } \Omega, \\ \quad \partial_\nu p_i = 0 & \text{on } \partial\Omega, \end{cases} \tag{4.38}$$

where $v \in U$ is unitary, $\bar{y}_i = \Pi(u_i)$, and $p_i = \Pi'(u_i)v$ for $i = 1, 2$. Taking the difference of the first equations in (4.38) for $i = 1, 2$, testing with $y_1 - y_2$, and using the mean value theorem we get for some $C_f > 0$ that

$$C_f \|y_1 - y_2\|_H^2 \le \|\nabla y_1 - \nabla y_2\|_{L^2(\Omega)}^2 + \int_\Omega (f(x, y_1) - f(x, y_2))(y_1 - y_2)\, dx$$

$$= \int_\Omega (u_1 - u_2)(y_1 - y_2)\, dx \le \|u_1 - u_2\|_U \|y_1 - y_2\|_H,$$

which yields the Lipschitz property $\|y_1 - y_2\|_H \le \frac{1}{C_f} \|u_1 - u_1\|_U$.

In order to show the local Lipschitz continuity of Π', we need to further assume condition (4.17). Consider now the difference of the right-hand side equations for $i = 1, 2$ in (4.38). Using standard PDE arguments (see, e.g., [44, Theorem 4.7]) we find

$$\|p_1 - p_2\|_{H^1(\Omega)} + \|p_1 - p_2\|_{C(\bar{\Omega})} \le C \|(\partial_y f(\cdot, \bar{y}_1) - \partial_y f(\cdot, \bar{y}_2))p_1\|_{L^2(\Omega)}$$

$$\le CL \|p_1\|_{C(\bar{\Omega})} \|\bar{y}_1 - \bar{y}_2\|_{L^2(\Omega)} \le C \frac{L}{C_f} c \|v\|_{L^2(\Omega)} \|u_1 - u_2\|_{L^2(\Omega)}.$$

Here, we also used the estimate $\|p_1\|_{C(\bar{\Omega})} \le c\|v\|_{L^2(\Omega)}$ from Theorem 4.7 for $u \in \mathcal{C}_{ad}$.

For the desired error bounds we focus now on the state equations

$$\begin{cases} -\Delta y + \mathcal{N}(x, y) = u & \text{in } \Omega, \\ \partial_\nu y = 0 & \text{on } \partial\Omega, \end{cases} \quad \text{and} \quad \begin{cases} -\Delta y + f(x, y) = u & \text{in } \Omega, \\ \partial_\nu y = 0 & \text{on } \partial\Omega, \end{cases} \tag{4.39}$$

and the associated adjoints

$$\begin{cases} -\Delta p + \partial_y \mathcal{N}(x, \bar{y})p = v & \text{in } \Omega, \\ \partial_\nu p = 0 & \text{on } \partial\Omega, \end{cases} \quad \text{and} \quad \begin{cases} -\Delta p + \partial_y f(x, \bar{y})p = v & \text{in } \Omega, \\ \partial_\nu p = 0 & \text{on } \partial\Omega. \end{cases} \tag{4.40}$$

The main approximation result is stated below. It guarantees that the uniform approximation properties of the control-to-state operator Π and its derivative (compare (2.2) and (2.7) of Theorem 2.3 and Assumption 2.4, respectively) are met by the corresponding learning-informed operators.

Proposition 4.12. *Let $M > K > 0$, with K being the constant from (4.11). Suppose the first inequality in (4.19) holds for f for every y such that $\|y\|_{L^\infty(\Omega)} \le K$. Assume that $\mathcal{N} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ satisfies the approximation property*

$$\sup_{\|y\|_{L^\infty(\Omega)} < M} \|f(\cdot, y) - \mathcal{N}(\cdot, y)\|_U \le \epsilon, \tag{4.41}$$

for $\epsilon > 0$ sufficiently small. Then, the following error estimate holds :

$$\|y_0 - y_\epsilon\|_H \le C\epsilon, \quad \text{for all } u \in \mathcal{C}_{ad}, \tag{4.42}$$

where the constant $C > 0$ depends only on f, and y_ϵ, y_0 are solutions of the left and right equations of (4.39) respectively. Moreover, assuming (4.17) and also that the condition

$$\sup_{\|y\|_{L^\infty(\Omega)} < M} \|\partial_y f(\cdot, y) - \partial_y \mathcal{N}(\cdot, y)\|_U \le \epsilon_1, \tag{4.43}$$

holds for sufficiently small $\epsilon_1 > 0$, then, there exist some constants $C_0 > 0$ and $C_1 > 0$ so that

$$\|p_0 - p_\epsilon\|_{H^1(\Omega) \cap C(\bar{\Omega})} \le C_1\epsilon_1 + C_0\epsilon, \quad \text{for all } u \in \mathcal{C}_{ad}, \tag{4.44}$$

where p_ϵ, p_0 are solutions of the left and right equations of (4.40) respectively.

Proof. Let y_ϵ and y_0 be solutions of the learning-informed PDE and the original PDE, respectively. Recall that the H^1 norms of both y_ϵ and y_0 are bounded by $K > 0$. Subtracting the two PDEs we get

$$-\Delta(y_0 - y_\epsilon) = \mathcal{N}(\cdot, y_\epsilon) - f(\cdot, y_0) \text{ in } \Omega \quad \text{and} \quad \partial_\nu(y_0 - y_\epsilon) = 0 \text{ on } \partial\Omega. \tag{4.45}$$

Using the same technique as in the proof of Theorem 4.7, the equation in (4.45) can be rewritten as

$$\left(-\Delta + \kappa_0 + (\partial_y f(\cdot, \zeta_\epsilon))^+\right)(y_0 - y_\epsilon) = \mathcal{N}(\cdot, y_\epsilon) - f(\cdot, y_\epsilon) + (\kappa_0 - (\partial_y f(\cdot, \zeta_\epsilon))^-)(y_0 - y_\epsilon), \qquad (4.46)$$

where ζ_ϵ is a pointwise convex combination of y_0 and y_ϵ that results from a pointwise application of the mean value theorem, and $\kappa_0 > 0$ is a fixed small constant. We have then the estimate

$$\frac{\kappa_0}{C_h} \|y_0 - y_\epsilon\|_{H^1(\Omega)} + \|y_0 - y_\epsilon\|_{C(\overline{\Omega})}$$
$$\leq (\kappa_0 + C_l)(\|\mathcal{N}(\cdot, y_\epsilon) - f(\cdot, y_\epsilon)\|_{L^2(\Omega)} + \left\|(\kappa_0 - (\partial_y f(\cdot, \zeta_\epsilon))^-)(y_0 - y_\epsilon)\right\|_{L^2(\Omega)}),$$

Rearranging the above inequality, and taking into account the Lipschitz continuity of $\partial_y f$ and the condition (4.19) for ζ_ϵ for which it holds $\|\zeta_\epsilon\|_{L^\infty(\Omega)} \leq K$, for sufficiently small ϵ we derive finally

$$\|y_0 - y_\epsilon\|_H \leq C\epsilon.$$

For deriving (4.44) we use a similar approach. Let p_ϵ and p_0 be the solutions of the left and right equations in (4.40), respectively. Subtracting these two equations gives

$$-\Delta(p_\epsilon - p_0) + \partial_y f(x, y_0)(p_\epsilon - p_0) = (\partial_y f(x, y_0) - \partial_y \mathcal{N}(x, y_\epsilon))p_\epsilon \quad \text{in } \Omega,$$
$$\partial_\nu(p_\epsilon - p_0) = 0 \quad \text{on } \partial\Omega. \qquad (4.47)$$

Using again the same trick as above, we rewrite (4.47) as

$$-\Delta(p_\epsilon - p_0) + (\kappa_1 + (\partial_y f(x, y_0))^+)(p_\epsilon - p_0)$$
$$= (\partial_y f(x, y_0) - \partial_y \mathcal{N}(x, y_\epsilon))p_\epsilon + (\kappa_1 - (\partial_y f(x, y_0))^-)(p_\epsilon - p_0), \qquad (4.48)$$

and then similarly we get

$$\|p_\epsilon - p_0\|_{H^1(\Omega)} \leq C \|p_\epsilon\|_{C(\overline{\Omega})} \|\partial_y f(\cdot, y_0) - \partial_y \mathcal{N}(\cdot, y_\epsilon)\|_{L^2(\Omega)}, \qquad (4.49)$$

for some constant C independent of both p_0 and p_ϵ, but depending on the constants C_h and C_l. The estimate in (4.49) holds also for $\|p_\epsilon - p_0\|_{C(\overline{\Omega})}$ but with a different constant, say $\tilde{C} > 0$. Focusing on the right-hand side of the inequality above and using the triangle inequality we have

$$\|\partial_y f(\cdot, y_0) - \partial_y \mathcal{N}(\cdot, y_\epsilon)\|_{L^2(\Omega)} \leq \|\partial_y f(\cdot, y_0) - \partial_y f(\cdot, y_\epsilon)\|_{L^2(\Omega)}$$
$$+ \|\partial_y f(\cdot, y_\epsilon) - \partial_y \mathcal{N}(\cdot, y_\epsilon)\|_{L^2(\Omega)} \leq L \|y_0 - y_\epsilon\|_{L^2(\Omega)} + \epsilon_1,$$

where L is the local Lipschitz constant of $\partial_y f(\cdot, \cdot)$ for those $y \in H^1(\Omega) \cap C(\overline{\Omega})$ with $\|y\|_{L^\infty(\Omega)} \leq K$.

Finally we need to estimate $\|p_\epsilon\|_{C(\overline{\Omega})}$ in (4.49). For this we note that for sufficiently small ϵ_1, the second bound in (4.21) also holds for the solution of PDEs with $\mathcal{N}$. This yields the estimate

$$\|p_\epsilon\|_{C(\overline{\Omega})} \leq C_c \|v\|_{L^2(\Omega)}, \qquad (4.50)$$

with the constant C_c independent of v and ϵ. Finally we conclude

$$\|p_0 - p_\epsilon\|_{H^1(\Omega) \cap C(\overline{\Omega})} = \sup_{\|v\|_{L^2(\Omega)} \leq 1} \|p_0 - p_\epsilon\|_{H^1(\Omega) \cap C(\overline{\Omega})}$$
$$= \sup_{\|v\|_{L^2(\Omega)} \leq 1} \|p_0 - p_\epsilon\|_{H^1(\Omega)} + \|p_0 - p_\epsilon\|_{C(\overline{\Omega})}$$
$$\leq C_c(C + \tilde{C})(L\epsilon + \epsilon_1) \leq C_1 \epsilon_1 + C_0 \epsilon,$$

which ends the proof. $\qquad \square$

Remark 4.13. *Notice that the condition (4.19) imposed on all y with $\|y\|_{L^\infty(\Omega)} \leq K$ in fact enforces a unique solution to the semilinear PDE (4.2), which also satisfies the same constraint. It is possible to treat the multi-solution case using a similar strategy as Theorem 4.7, by using Γ–convergence arguments to show the convergence of $y_\epsilon \to y$ in a certain sense, and then apply the condition (4.19) to y_0.*

Remark 4.14. *The results above also hold for more general types of boundary conditions, including homogeneous Dirichlet boundary conditions.*

4.4 Existence of solutions of the learning-informed optimal control

After having replaced the unknown f by the neural network based approximation $\mathcal{N}$ we are now interested in the following optimal control problem with a partially learning-informed state equation:

$$\text{minimize} \quad J(y,u) := \frac{1}{2}\|y - g\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\|u\|^2_{L^2(\Omega)}, \quad \text{over } (y,u) \in H^1(\Omega) \times L^2(\Omega), \tag{4.51}$$

$$\text{s.t.} \quad -\Delta y + \mathcal{N}(x,y) = u \quad \text{in } \Omega, \quad \partial_\nu y = 0 \text{ on } \partial\Omega, \tag{4.52}$$

$$u \in \mathcal{C}_{ad}. \tag{4.53}$$

In what follows we prove the existence of an optimal control for the problem (4.51)–(4.53). Here we consider that the control-to-state operator is single-valued, that is, the learning-informed PDE (4.52) has a unique solution for every $u \in \mathcal{C}_{ad}$. According to Proposition 2.1, we only need to check that the operator $Q_{\mathcal{N}} : U \to H$ is weakly sequentially closed. In fact, an even stronger property holds true as we show next.

Proposition 4.15. *Let $\mathcal{N} \in C^\infty(\mathbb{R}^d \times \mathbb{R})$ be a neural network such that any solution of the learning-informed PDE (4.52) satisfies a bound as in (4.11). Then the reduced operator $Q_{\mathcal{N}} : U = L^2(\Omega) \supset C_{ad} \to H = L^2(\Omega)$ induced from the control-to-state map of (4.52) is weakly-strongly continuous, in the sense that if $u_n \rightharpoonup u$ in U and $y_n \in \Pi_{\mathcal{N}}(u_n)$ then, $y_n \to y$ in H for some $y \in \Pi_{\mathcal{N}}(u)$.*

Proof. For $u_n \rightharpoonup u$ in $U \cap \mathcal{C}_{ad}$ and $y_n \in \Pi_{\mathcal{N}}(u_n)$, $(y_n)_{n\in\mathbb{N}}$ is a bounded sequence in $Y = H^1(\Omega) \cap C(\overline{\Omega})$ as $(u_n)_{n\in\mathbb{N}} \subset \mathcal{C}_{ad}$ is a bounded set in $L^\infty(\Omega)$. Thus, up to a subsequence, still denoted by (y_n), there is $y \in H^1(\Omega)$ such that $y_n \rightharpoonup y$ in $H^1(\Omega)$. Since $H^1(\Omega)$ embeds compactly into H, we can consider that $y_n \to y$ strongly in H. We show that $y = \Pi_{\mathcal{N}}(u)$, i.e., y is a weak solution of the PDE in (4.52). Since y_n is the weak solution of (4.52) with right hand-side u_n, we have

$$\int_\Omega \nabla y_n \cdot \nabla v \, dx + \int_\Omega \mathcal{N}(x, y_n) v \, dx = \int_\Omega u_n v \, dx \quad \text{for all } v \in H^1(\Omega). \tag{4.54}$$

We only need to show that

$$\int_\Omega (\mathcal{N}(x, y_n) - \mathcal{N}(x, y)) \, v \, dx = 0, \tag{4.55}$$

since the convergence of the other two terms readily follows from weak convergence. Taking into account that $\mathcal{N} \in C^1(\mathbb{R}^d \times \mathbb{R})$ we have that for every $M > 0$, there exists an $L_M > 0$ such that for every $x \in \Omega$ and $y_1, y_2 \in [-M, M]$, we have

$$|\mathcal{N}(x, y_1) - \mathcal{N}(x, y_2)| \le L_M |y_1 - y_2|. \tag{4.56}$$

Using the estimate (4.11), we have that $(y_n)_{n\in\mathbb{N}}$ and, hence, y are uniformly bounded in $L^\infty(\Omega)$, say by a constant $M > 0$. Thus we have

$$\|\mathcal{N}(\cdot, y_n) - \mathcal{N}(\cdot, y)\|_U \le L_M \|y_n - y\|_H.$$

Due to the inequality above and the strong convergence of $y_n \to y$ in H, (4.55) is verified. Passing to the limit $n \to \infty$ in (4.54) we get that y is a weak solution of (4.52) corresponding to u. Since any other subsequence of $(y_n)_{n\in\mathbb{N}}$ will have a further subsequence that converges to $\Pi_{\mathcal{N}}(u)$ the assertion follows. $\square$

For the error analysis on the optimal controls of (4.51) with (4.52) to solutions from (4.1) with (4.2), we can readily apply Theorems 2.3, 2.6 and 2.5 for the monotone function f, in view of the error bounds shown in Proposition 4.12. For the nonmonotone case, these results are still applicable up to a selection of subsequences of the solutions.

Finally, we would like to make a remark regarding the approximation of $f : \Omega \times \mathbb{R} \to \mathbb{R}$ in a semilinear PDE, given a set of input-output data. The input data is a family of sampled points from $\Omega \times [y_{min}, y_{max}]$, denoted by $(x_i, y(x_i))_{i\in I}$, and the outputs are the corresponding values $(f(x_i, y(x_i)))_{i\in I}$, which are computed from (4.2) via

$$f(x_i, y(x_i)) = u(x_i) + \Delta y(x_i).$$

In real world applications, we assume that we have access to the data points $y(x_i)$ and thus also to $\Delta y(x_i)$, while u is a control which is at our disposal to be tuned. In order to be consistent with the functional analytic setting, one needs to give pointwise meaning to Δy, which in general is an object in $H^{-1}(\Omega)$, only. This can be achieved by choosing controls $u \in \mathcal{C}_{ad}$ of sufficient regularity. Indeed, since both f and y are continuous functions when choosing continuous u, equation (4.2) implies that Δy is continuous, too, and hence admits a pointwise evaluation.

4.5 Numerical algorithm for the optimal control problems

In this section we briefly describe an algorithm for solving the optimal control problem (4.1). Even though it is suitable for rather general problems, we outline it here for the version with the learning-informed state equation.

In order to compute a numerical solution, we first state the Karush-Kuhn-Tucker (KKT) conditions, which are justified by constraint regularity (see [45] for a general setting):

$$
\begin{aligned}
-\Delta y + \mathcal{N}(\cdot, y) - u = 0 \ \ \text{in } \Omega, \quad \partial_\nu y = 0 \ \ \text{on } \partial\Omega, \\
-\Delta p + \partial_y \mathcal{N}(\cdot, y)p + y = g \ \ \text{in } \Omega, \quad \partial_\nu p = 0 \ \ \text{on } \partial\Omega, \\
-p + \lambda + \alpha u = 0 \ \ \text{in } \Omega, \\
\lambda - \max(0, \lambda + c(u - \overline{u})) - \min(0, \lambda + c(u - \underline{u})) = 0 \ \ \text{in } \Omega,
\end{aligned}
\tag{4.57}
$$

where $c > 0$ is some constant, which in practice, is useful to be chosen $c = \alpha$. The first equation with its boundary condition is just the learning-informed PDE constraint, while the next one is the associated adjoint equation. The third equation represents optimality w.r.t. u and, together with the last one, it incorporates the control constraint $\underline{u} \le u \le \overline{u}$. Indeed, notice that the last equation is equivalent to the usual complementarity system as it secures a.e. that

$$
\lambda = 0 : \underline{u} < u < \overline{u}, \quad \lambda \ge 0 : u = \underline{u}, \quad \lambda \le 0 : u = \overline{u}.
$$

Letting $\phi := (y, u, p, \lambda)^\top$, (4.57) can be compactly rewritten as the nonsmooth equation

$$
M_\mathcal{N}(\phi) - (0, g, 0, 0)^\top = 0.
\tag{4.58}
$$

For solving (4.58), we employ a semi-smooth Newton method (SSN); see, e.g., [25]. It operates as follows: Given an initial guess ϕ_0 of a solution to (4.58), compute for all $k = 0, 1, 2, \dots$

$$
\phi_{k+1} = \phi_k - (\mathcal{G}_\mathcal{N}(\phi_k))^{-1}(M_\mathcal{N}(\phi_k) - (0, g, 0, 0)^\top).
$$

Here, $\mathcal{G}_\mathcal{N}(\phi_k)$ is a Newton derivative of the operator $M_\mathcal{N}$ at ϕ_k given by

$$
\mathcal{G}_\mathcal{N}(\phi_k) =
\begin{pmatrix}
-\Delta + \partial_y \mathcal{N}(\cdot, y_k) & 0 & -\text{Id} & 0 \\
\partial_{yy}\mathcal{N}(\cdot, y_k)p_k + \text{Id} & -\Delta + \partial_y \mathcal{N}(\cdot, y_k) & 0 & 0 \\
0 & -\text{Id} & \alpha\,\text{Id} & \text{Id} \\
0 & 0 & -cG_k & \text{Id} - G_k
\end{pmatrix},
$$

where for $x \in \Omega$,

$$
G_k(x) :=
\begin{cases}
0, & \text{if } c(\underline{u}(x) - u_k(x)) \le \lambda_k(x) \le c(\overline{u}(x) - u_k(x)), \\
1, & \text{else},
\end{cases}
$$

is a Newton derivative that corresponds to the nonsmooth functions $\max(0, \cdot)$ and $\min(0, \cdot)$ in (4.57). SSN can be shown to converge locally at a superlinear rate, provided ϕ_0 is sufficiently close to a solution and the selection of Newton derivatives for $M_\mathcal{N}$ is uniformly bounded and invertible along the iteration sequence; see [25] and [27]. Moreover, under a nondegeneracy assumption the method exhibits a mesh independent convergence upon proper discretization of (4.58); see [24, 27]. Globalization of the SSN iteration can be achieved, e.g., by employing a path search [15, 40], which we did not pursue here, however. Rather we intertwined SSN with a sequential quadratic programming (SQP) iteration, with the latter specified below. This combination helped the globally convergent SQP solver to escape from unfavorable local minimizers or stationary points. Obviously, one cannot expect a general theoretical result supporting such a behavior. It, hence, merely reflects a useful numerical observation, in particular in connection with our example with a nonmonotone f.

SQP algorithm Here we consider the reduced SQP approach which operates on the reduced optimal control problem. Given an estimate u_k of an optimal control, in every iteration it seeks to solve the following quadratic problem:

$$
\begin{aligned}
\text{minimize} \quad & \langle \mathcal{J}'_\mathcal{N}(u_k) + \tfrac{1}{2} H_k(u_k)\delta_u, \delta_u \rangle_{U^*, U}, \quad \text{over } \delta_u \in U, \\
\text{subject to} \quad & \underline{u} \le u_k + \delta_u \le \overline{u} \quad \text{a.e. in } \Omega,
\end{aligned}
\tag{4.59}
$$

where $\mathcal{J}'_{\mathcal{N}}(u_k)$ is the Fréchet derivative of the reduced functional $\mathcal{J}_{\mathcal{N}}$, and $H_k(u_k)$ is a positive definite approximation of the second-order derivative of $\mathcal{J}_{\mathcal{N}}$ at u_k. First-order optimality for (4.59) yields

$$
\begin{aligned}
&\mathcal{J}'_{\mathcal{N}}(u_k) + H_k(u_k)\delta_u + \lambda = 0, \\
&\lambda - \max(0, \lambda + c(u_k + \delta_u - \overline{u})) - \min(0, \lambda + c(u_k + \delta_u - \underline{u})) = 0,
\end{aligned} \tag{4.60}
$$

for some fixed $c > 0$. This nonsmooth system can be again solved using a semi-smooth Newton method which yields $\delta_{u,k}$ and λ_k. Concerning the Hessian approximation, in our implementation we choose $H_k(u_k) := (\mathcal{J}'_{\mathcal{N}}(u_k))^* \mathcal{J}'_{\mathcal{N}}(u_k)$, where '*' denotes the adjoint operator.

For globalization we use a classical line search with the merit function

$$
\Phi_k(\mu) = \mathcal{J}_{\mathcal{N}}(u_k + \mu\delta_{u,k}) + \beta_k \Psi_k(\mu) \quad \text{for some } \beta_k > 0, \tag{4.61}
$$

where

$$
\Psi_k(\mu) := \left\| (u_k + \mu\delta_{u,k} - \overline{u})^+ \right\|_{L^2(\Omega)} + \left\| (u_k + \mu\delta_{u,k} - \underline{u})^- \right\|_{L^2(\Omega)},
$$

with $a^+ := \max\{a, 0\}$, and $a^- := \min\{0, a\}$. We employ a backtracking line search method starting with $\mu := 1$ to decide on the step length. Note that the reduced problem requires to enforce the PDE constraint for every u_k. For this purpose a (smooth) Newton iteration was embedded into every SQP update step. This Newton iteration is terminated when $\| -\Delta_h y_k + \mathcal{N}(\cdot, y_k) - u_k \|_{H^{-1}(\Omega)} \leq \text{tol} = 10^{-16}$ or a maximum of 15 iterations was reached.

To summarize, we utilize the following overall algorithm:

Algorithm 1 A semi-smooth Newton SQP algorithm for PDE control problems

---

- Initialization: Choose $\phi_0 := (y_0, u_0, p_0, \lambda_0)$, and compute $\Phi_0(0)$. Fix a lower bound $\epsilon > 0$ for the step length, choose $\rho \in (0, 1)$, and $\beta_0 > 0$. Set $k := 0$.

- Unless the stopping criteria are satisfied, iterate:

  (1) Compute an update direction $\delta_{u,k}$ by solving (4.60) using SSN. Let $\mu_k^0 := 1$, $y_k^{-1} := y_k$ and set $l := 0$. Iterate:

  (a1) Compute $y_k^l := \Pi_{\mathcal{N}}(u_k + \mu_k^l \delta_{u,k})$, where $\Pi_{\mathcal{N}}$ is realized by performing Newton iterations as a nonlinear PDE solver initialized by y_k^{l-1}.
  Setting $y := y_k^l$ and $u := u_k + \mu_k^l \delta_{u,k}$ compute the remaining quantities in ϕ_k^l according to (4.57) with $p =: p_k^l$ and $\lambda =: \lambda_k^l$. This yields ϕ_k^l.

  (a2) Increase β_k, if necessary, to get β_k^l.

  (a3) Check the Armijo condition (4.63).
  If it is satisfied, then set $l_k := l$ and continue with step (2); otherwise update $\mu_k^{l+1} := r\mu_k^l$, $l := l + 1$.
  If $\mu_k^{l+1} < \epsilon$, then terminate the algorithm; otherwise return to Step (a1).

  (2) Set $\phi_{k+1} := \phi_k^{l_k}$, and $\beta_{k+1} := \beta_k^{l_k}$, and $k := k + 1$.

- Output: The value of ϕ_k which contains both the control and state variables.

In our examples, we choose $\mu_0 = 1$, $\epsilon = 10^{-5}$, $r = 2/3$, and $\beta_0 = \|\lambda_0\|_{L^2(\Omega)} + 1$. In order to solve the nonsmooth system in (4.60), we employ a primal-dual active set strategy (pdAS), which was shown to be equivalent to an efficient SSN solver for classes of constrained optimization problems [25]. For the precise set-up of pdAS and the associated active/inactive set estimation we also refer to [25]. For minimizing quadratic objectives subject to box constraints and utilizing highly accurate linear system solvers, pdAS is typically terminated when two consecutive active and inactive set estimates coincide. We recall here that the active set for (4.59) at the solution $\delta_{u,k}$ is a subset $\mathcal{A}_k$ of Ω with $(u_k + \delta_{u,k})(x) \in [\underline{u}(x), \overline{u}(x)]$ for $x \in \mathcal{A}_k$; $\mathcal{I}_k := \Omega \setminus \mathcal{A}_k$ denotes the associated inactive set. Alternatively one may stop the iteration once the residual norm of the nonsmooth system at an iterate drops below a user specified tolerance. In view of (4.60) and constraint satisfaction, the function $\Psi_k(\mu)$ in (4.61) appears irrelevant as a penalty for violations of the box constraints. However, it becomes relevant when early stopping is employed in SSN (respectively pdAS). In this case we still need to guarantee that $\delta_{u,k}$ is a descent direction for our

merit function to obtain sufficient decrease of Φ_k in our line search (4.63). This is needed for getting convergence of (u_k) (along a subsequence) to a stationary point. For deriving a proper stopping rule for SSN to guarantee sufficient decrease, we multiply the first equation in (4.60) by the solution δ_u, use $\lambda(u_k + \delta_u - \overline{u})(u_k + \delta_u - \underline{u}) = 0$ a.e. in Ω and the feasibility of $u_k + \delta_u$, both according to the second line in (4.60). We further set $\beta_k > \|\lambda\|_U$ (upon identifying $U^* \widehat{=} U$) to find

$$\langle \mathcal{J}'_{\mathcal{N}}(u_k), \delta_u \rangle_{U^*,U} + \beta_k (\underbrace{\Psi_k(1)}_{=0} - \Psi_k(0)) \leq -\langle H_k(u_k)\delta_u, \delta_u \rangle_{U^*,U} < 0,$$

unless $\delta_u = 0$, i.e., u_k is stationary for the original reduced problem. Here, δ_u replaces $\delta_{u,k}$ in $\Psi_k(1)$. This motivates our termination rule for SSN when solving (4.60). In fact, let superscript l denote the iteration index of SSN for the outer iteration k, i.e., for given u_k. For some initial guess (δ_u^0, λ^0) (typically chosen to be $(\delta_{u,k-1}, \lambda_{k-1})$) SSN computes iterates (δ_u^l, λ^l), $l \in \mathbb{N}$, and terminates at iteration l_k, which is the smallest index with

$$\begin{aligned} &\langle \mathcal{J}'_{\mathcal{N}}(u_k), \delta_u^{l_k} \rangle_{U^*,U} + \beta_k(\Psi_k(1) - \Psi_k(0)) \leq -\xi \langle H_k(u_k)\delta_u^{l_k}, \delta_u^{l_k} \rangle_{U^*,U} \\ &\text{and} \quad \Psi_k(1) \leq (1-\xi)\Psi_k(0) \end{aligned} \tag{4.62}$$

for some $\xi \in (0,1)$, with $\beta_k > \|\lambda^{l_k}\|_U$, and where $\delta_u^{l_k}$ is used in $\Psi_k(1)$. In our tests, we choose $\xi = 0.9$, and terminate SSN iterations whenever (4.62) is satisfied or two consecutive active set estimates are identical. Then we set $\delta_{u,k} := \delta_u^{l_k}$, $\lambda_k := \lambda^{l_k}$, and determine a suitable step size μ_k.

For the latter we use a backtracking line search based on the Armijo condition [38]. Indeed, given u_k, $\delta_{u,k}$, and λ_k, let l now denote the running index of the line search iteration. Then $l_k \in \mathbb{N}$ is the smallest index such that

$$\Phi_k(\mu_k^{l_k}) - \Phi_k(0) \leq \kappa \mu_k^{l_k} \left(\langle \mathcal{J}'_{\mathcal{N}}(u_k), \delta_{u,k} \rangle_{U^*,U} + \beta_k(\Psi_k(1) - \Psi_k(0)) \right), \tag{4.63}$$

for some parameter $0 < \kappa < 1$, and $\beta_k = \max\{\beta_{k-1}, \zeta\|\lambda_k\|_U\} > \|\lambda_k\|_U$, for some $\zeta > 1$ in (a2). In our implementation we use $\kappa = 10^{-3}$ and $\zeta = 2$.

Regarding the stopping criteria for the SQP iterations, we set a tolerance for the norm of the residual of (4.57) along with a maximal number of iterations. We note here that (4.57) matches (4.60) upon introducing the adjoint state for efficiently computing $\mathcal{J}'_{\mathcal{N}}(u_k)$ to the latter.

In our implementation we simplified the Newton derivative of the first-order system (4.57) by dropping the second-order derivatives $\partial_{yy}\mathcal{N}(\cdot, y_k)p_k$ from $\mathcal{G}_{\mathcal{N}}(\phi_k)$. The corresponding approximation reads

$$\begin{pmatrix} -\Delta + \partial_y \mathcal{N}(\cdot, y_k) & 0 & -\text{Id} & 0 \\ \text{Id} & -\Delta + \partial_y \mathcal{N}(\cdot, y_k) & 0 & 0 \\ 0 & -\text{Id} & \alpha\text{Id} & \text{Id} \\ 0 & 0 & -cG_k & \text{Id} - G_k \end{pmatrix} \simeq \mathcal{G}_{\mathcal{N}}(\phi_k).$$

This helped to stabilize the SSN iterations, while maintaining almost the same convergence rates as for the exact Newton derivative in our tests.

4.6 Numerical results on distributed optimal control of semilinear elliptic PDEs

Our first test problem is given by

$$\left. \begin{aligned} \text{minimize} \quad & \frac{1}{2}\|y - g\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\|u\|^2_{L^2(\Omega)}, \text{ over } (y, u) \in H^1(\Omega) \times L^2(\Omega), \\ \text{subject to} \quad & -\Delta y + f(x, y) = u \text{ in } \Omega := (0, 2) \times (0, 2), \quad \partial_\nu y = 0 \text{ on } \partial\Omega, \\ & -20 \leq u \leq 20. \end{aligned} \right\} \tag{4.64}$$

with exact underlying nonlinearity $f(x, z) = z + 5\cos^2(\pi x_1 x_2)z^3$ and $x = (x_1, x_2) \in \mathbb{R}^2$, $z \in \mathbb{R}$.

4.6.1 Training of artificial neural networks

For learning the function f we use neural networks that are built from standard (multi-layer) feed-forward networks. Their respective architecture together with the loss function as well as the training data and method are specified next.

Loss function and training method   Let $\Theta = (W, b)$ denote the parameters associated with an ANN $\mathcal{N} =: \mathcal{N}_\Theta$ that needs to be trained by solving an associated minimization problem; compare (3.2). We use here the mean squared error

$$\mathfrak{d}(\mathbf{x}, \mathbf{f}) = \frac{1}{n_D} \sum_{j=1}^{n_D} |\mathcal{N}_\Theta(\mathbf{x}_j) - \mathbf{f}_j|^2,$$

as a loss function, no regularization, i.e, $\mathfrak{r} \equiv 0$, and $\mathcal{F}_{\mathrm{ad}}$ is the full space. In this context, $(\mathbf{x}_j, \mathbf{f}_j)_{j=1}^{n_D}$ are the input-output training pairs. For simplicity of presentation we assume that n_D is larger than the number of unknowns in Θ.

For solving (3.2), we adopt a Bayesian regularization method [33] which is based on a Levenberg-Marquardt (LM) algorithm, and is available in MATLAB packages. We initialized the LM algorithm by unitary random vectors using the Nguyen-Widrow method [36], and terminated it as soon as the Euclidean norm of the gradient of the loss function dropped below 10^{-7} or a maximum of 1000 iterations was reached. For other methods that are suitable for this task we refer to the overview in [10].

Architecture of the network   In order to have a representative study of the influence of ANN architectures on our computational results, we used networks with a total number of hidden layers (HL) equal to 1, 3 or 5. In each choice, we further varied the number of neurons per layer such that the final number of unknowns in Θ (degree(s) of freedom; DoF) remained in essence the same. Such tests were performed for three different DoF (small, medium, large) resulting in a total of nine different architectures; cf. Table 4.1. All underlying networks operate with input layer size of three neurons and one neuron in the output layer. In all tests for this example, the log-sigmoid transfer function (`logsig` in MATLAB) was chosen as the activation function at all the hidden layers.

	HL 1	HL 2	HL 3	HL 4	HL 5	Total DoF
	Small DoF					
No. of neurons	30	-	-	-	-	151
No. of neurons	6	10	5	-	-	155
No. of neurons	3	5	10	5	1	155
	Medium DoF					
No. of neurons	60	-	-	-	-	301
No. of neurons	10	12	10	-	-	313
No. of neurons	5	8	10	8	6	307
	Large DoF					
No. of neurons	120	-	-	-	-	601
No. of neurons	15	18	13	-	-	609
No. of neurons	10	10	15	10	10	596

Table 4.1: Architecture of networks. HL i: i hidden layers; DoF: degrees of freedom in Θ.

Training and validation data   The training data rest on chosen control actions $(u_j)_{j=1}^{n_D} \subset \mathcal{C}_{ad}$ with

$$u^j = -2d_j \pi^2 \cos(\pi x_1) \cos(\pi x_2)$$
$$- d_j \cos(\pi x_1) \cos(\pi x_2) - 5d_j^3 \cos^2(\pi x_1 x_2) \cos^3(\pi x_1) \cos^3(\pi x_2),$$

and $(d_j) = \{[0.01 : 0.4 : 2.01]\}$ (in MATLAB notation). The procedure for generating the training data is as follows: First, numerical solutions are computed on a uniform discrete mesh $\Omega_h = \{x^k\}_{k=1}^{\bar{N}_h}$ (represented here by the associated mesh nodes including those on $\partial\Omega$) with mesh width $h = \frac{1}{50}$, and $\bar{N}_h = (n_h + 1)^2$, $n_h = 1/h$. The Laplace operator is discretized by the standard five-point finite difference stencil respecting the homogeneous Neumann boundary conditions. This yields the $N_h \times N_h$-matrix Δ_h related to nodes x_k in Ω with $N_h = (n_h - 1)^2$. The nonlinearity as well as the controls are evaluated at such mesh points x^k, and the resulting discrete nonlinear PDE (4.64) is solved by Newton's method. The Newton iteration is terminated once the PDE residual in the discrete $H^{-1}(\Omega)$-norm drops below 10^{-16}, or a maximum of 30

iterations is reached. Thus for each u^j, $j = 1, \ldots, n_D$, we obtain numerical values $y_h^j = (y_{h,1}^j, \ldots, y_{h,N_h}^j)^\top$ associated with the (interior) mesh nodes x^k and approximating $y^j(x^k) = -d_j \cos(\pi x_1^k) \cos(\pi x_2^k)$, the analytical PDE solution. Using these data we compute the output values of f denoted by $f_h^j \in \mathbb{R}^{N_h}$ according to the PDE by

$$f(x^k, y^j(x^k)) \approx u^j(x^k) + (\Delta_h y_h^j)_k =: f_{h,k}^j, \quad k = 1, \ldots, N_h, \quad j = 1, \ldots, n_D.$$

These input-output pairs both are prepossessed using `mapminmax` function in MATLAB without change of notation here. The training data are then obtained through subsampling $f_{h,k}^j$ by restriction to a coarse mesh Ω_H, with $H > h$. For this purpose we use $H \in \{0.2, 0.1, 0.08\}$ giving rise to a small, medium and large training set, respectively. The corresponding reduction rates are $1/10$, $1/5$, and $1/4$ with respect to the data for $h = 1/50$.

This subsampled data set is then split into a training data set, a validation data set and a testing data set at the ratio of $8 : 1 : 1$. In our tests, such a data partitioning is done randomly by using MATLAB's `randperm` function.

4.6.2 Numerical results

We start by comparing the exact, numerical and learning-based solutions, respectively. The exact reference solution is chosen to be

$$y^* = 1.5 \cos(\pi x_1) \cos(\pi x_2),$$

and the numerical approximation y_h resulted from a mesh with $h = 2^{-7}$ and the use of the exact nonlinearity f. The same grid is used for obtaining the numerical approximation of $y_\mathcal{N}$. Note, however, that the grid for data generation is different from the grid for numerical computation.

Our report on the experiments involves several discrete norms. In fact, for $z_h \in \mathbb{R}^{N_h}$ we have

$$|z_h|_1^2 := h^2 (\Delta_h z_h)^\top z_h, \quad \|z_h\|_0^2 := h^2 z_h^\top z_h,$$

where $|\cdot|_1$ and $\|\cdot\|_0$ correspond to the H^1-seminorm and L^2-norm, respectively.

	$|y_\mathcal{N} - y_h^*|_1$	$|y_\mathcal{N} - y_h^*|_1$	$|y_\mathcal{N} - y^*|_1$	$|y_\mathcal{N} - y^*|_1$	$\|y_\mathcal{N} - y_h^*\|_0$	$\|y_\mathcal{N} - y_h^*\|_0$	$\|y_\mathcal{N} - y^*\|_0$	$\|y_\mathcal{N} - y^*\|_0$
	min	max	min	max	min	max	min	max
1-L	0.2506	0.6532	0.2868	0.6713	0.0752	0.2422	0.0808	0.2435
3-L	0.2575	0.7537	0.2391	0.7777	0.0817	0.2524	0.0791	0.2565
5-L	0.2157	36.2640	0.2235	36.2731	0.0539	29.4926	0.0544	29.4936
	mean	deviation	mean	deviation	mean	deviation	mean	deviation
1-L	0.4276	0.1099	0.4496	0.1075	0.1472	0.0484	0.1506	0.0485
3-L	0.3853	0.1350	0.4003	0.1687	0.1425	0.0462	0.1268	0.0482
5-L	3.0242	8.9087	3.0287	8.9103	2.1309	7.3143	2.1299	7.3149

Table 4.2: Statistics on learning-informed PDEs with different layers in neural networks using small size training data, small DoF in Θ, and 15 samples in total.

Table 4.2 depicts the approximation results for different ANN architectures with small DoF as described in 4.1 and in all cases the small training data set.

We find that the 1-layer network is robust in terms of the statistical quantities shown, and the 3-layer network has the smallest errors on average, but exhibits a larger deviation than the 1-layer network. The 5-layer network yields the smallest error, but also the largest ones with a very big deviation. This behavior may be attributed to the fact that deeper networks give rise to increasingly more nonlinear compositions entering the loss function. This may be stabilized by tuned initializations, additional regularization, or sufficient training data. A study along these lines, however, is not within the scope of the present work as noted earlier.

In Table 4.3, we provide statistics on the influence of the number of neurons for fixed layers. We use 3-layer networks and medium sized training data for this set of experiments. All three levels of DoF for the networks as given in Table 4.1 are studied. The results in terms of 'mean' and 'deviation' indicate that a large number of neurons gives typically better approximations when compared to the smaller size of DoFs. However, we also observe that the deviation and the maximum error increases with the number of DoF. This can be attributed to an increase in training error for increasing DoFs.

	$\lvert y_N - y_h^*\rvert_1$	$\lvert y_N - y_h^*\rvert_1$	$\lvert y_N - y^*\rvert_1$	$\lvert y_N - y^*\rvert_1$	$\lVert y_N - y_h^*\rVert_0$	$\lVert y_N - y_h^*\rVert_0$	$\lVert y_N - y^*\rVert_0$	$\lVert y_N - y^*\rVert_0$
	min	max	min	max	min	max	min	max
3-L S	0.0546	0.1658	0.0889	0.2211	0.0086	0.0546	0.0207	0.0515
3-L M	0.0090	0.1508	0.0876	0.2039	0.0026	0.0492	0.0168	0.0591
3-L L	0.0155	0.2815	0.0833	0.3306	0.0036	0.0901	0.0161	0.0996
	mean	deviation	mean	deviation	mean	deviation	mean	deviation
3-L S	0.1103	0.0357	0.1464	0.0329	0.0266	0.0125	0.0339	0.0095
3-L M	0.0631	0.0407	0.1113	0.0367	0.0170	0.0120	0.0250	0.0117
3-L L	0.0559	0.0626	0.1115	0.0609	0.0149	0.0205	0.0250	0.0204

Table 4.3: Statistics on learning-informed PDEs with different numbers of neurons in networks using medium size training data of 15 samples in total.

Next we present some computational results where we use the learning-informed PDE as constraint when numerically solving the optimal control problem (4.51). Here we consider a target function $g = y^* + \delta$ where δ is a variable denoting zero-mean Gaussian noise of standard deviation $\hat{\sigma}$, for different values of $\hat{\sigma}$. For convenience of comparison, we take y^* to be the solution from the last set of experiments. We denote by u_N and $\bar{u}$ the optimal controls with respect to the learning-informed PDE constraint and the original PDE constraint, respectively, both computed by the semi-smooth Newton algorithm as described in Section 4.5 with a fixed number of 30 iterations which turns out to be sufficient for this example, as the sum of all residual norms of the first-order system (4.57) is less than 10^{-10}. As before, y_N and $\bar{y}$ are the states corresponding to u_N and $\bar{u}$, respectively.

	Small DoF			Medium DoF			Large DoF		
	$\lVert u_N - \bar{u}\rVert_0$	$\lVert y_N - \bar{y}\rVert_0$	$\lvert y_N - \bar{y}\rvert_1$	$\lVert u_N - \bar{u}\rVert_0$	$\lVert y_N - \bar{y}\rVert_0$	$\lvert y_N - \bar{y}\rvert_1$	$\lVert u_N - \bar{u}\rVert_0$	$\lVert y_N - \bar{y}\rVert_0$	$\lvert y_N - \bar{y}\rvert_1$
	Small size of training data								
1-L	0.5578	0.0330	0.1609	0.3055	0.0283	0.1423	0.2548	0.0194	0.1143
3-L	0.3426	0.0274	0.1246	0.3597	0.0343	0.1777	**0.3932**	**0.0354**	**0.1722**
5-L	0.3888	0.0183	0.1041	0.1771	0.0117	0.0666	0.3986	0.0359	0.1698
	Medium size of training data								
1-L	0.2145	0.0071	0.0413	0.1153	0.0072	0.0587	0.0655	0.0029	0.0244
3-L	0.1647	0.0069	0.0419	0.0985	0.0082	0.0423	**0.0623**	**0.0046**	**0.0287**
5-L	0.2971	0.0271	0.1223	0.0325	0.0014	0.0081	0.0736	0.0064	0.0414
	Large size of training data								
1-L	0.1417	0.0089	0.0481	0.0920	0.0040	0.0266	0.0447	0.0009	0.0055
3-L	0.0566	0.0020	0.0126	0.0467	0.0024	0.0122	**0.0076**	**0.0004**	**0.0020**
5-L	0.1239	0.0070	0.0435	0.2135	0.0098	0.0645	0.0192	0.0018	0.0115

Using the same noisy data g (Gaussian noise of mean zero and deviation 0.1) with $\alpha = 0.001$ in all the tests

Table 4.4: Optimal control with learning-informed PDEs using different layers, different size of networks, and a variety of training data.

In general, we observe in Table 4.4 that most combinations give similar results. This shows the robustness of our proposed method with respect to a wide range of network architectures. Here, the presented errors are just computed from one specific initialization.

Note that when using 3-hidden-layer networks with large DoF, we observe a clear increase in the levels of accuracy for both the control and state variables as the training data increase from small to large size. These are highlighted with bold font numbers in Table 4.4. A similar behavior occurs for 1-hidden-layer and 5-hidden-layer networks. By fixing the 3-hidden-layer networks, and for each case of DoFs provided in Table 4.4, we are next interested in exploring how the noise level σ and the cost parameter α further influence the optimal control approximation.

From Table 4.5 we draw several interesting conclusions. In both, the noisy and noise free case, we have that the error $\lVert u_N - \bar{u}\rVert$ is proportional to the accuracy of the neural network approximation, and inverse proportional to $\sqrt{\alpha}$. This verifies the results of Theorem 2.6 and Theorem 2.5, respectively. The dependence on α could only be proved for the noise-free case in Theorem 2.6. Therefore the convergence rates provided by our tests here seem to indicate that better convergence rates or more relaxed assumptions appear plausible.

	Noise free			Mild noise $\hat{\sigma} = 0.05$			Larger noise $\hat{\sigma} = 0.5$		
	$\|u_\mathcal{N} - \bar{u}\|_0$	$\|y_\mathcal{N} - \bar{y}\|_0$	$|y_\mathcal{N} - \bar{y}|_1$	$\|u_\mathcal{N} - \bar{u}\|_0$	$\|y_\mathcal{N} - \bar{y}\|_0$	$|y_\mathcal{N} - \bar{y}|_1$	$\|u_\mathcal{N} - \bar{u}\|_0$	$\|y_\mathcal{N} - \bar{y}\|_0$	$|y_\mathcal{N} - \bar{y}|_1$
$\alpha = 0.00001$									
3-L-S NN	1.9523	0.0210	0.2041	1.9518	0.0210	0.2043	2.1480	0.0213	0.2085
3-L-M NN	0.1187	0.0018	0.0253	0.1190	0.0018	0.0253	0.1264	0.0018	0.0254
3-L-L NN	0.0213	0.0004	0.0046	0.0215	0.0004	0.0046	0.0258	0.0004	0.0047
$\alpha = 0.0001$									
3-L-S NN	1.3489	0.0395	0.2695	1.3560	0.0397	0.2705	1.4181	0.0410	0.2796
3-L-M NN	0.1361	0.0032	0.0314	0.1357	0.0032	0.0314	0.1384	0.0032	0.0315
3-L-L NN	0.0137	0.0005	0.0039	0.0136	0.0005	0.0039	0.0136	0.0005	0.0039
$\alpha = 0.001$									
3-L-S NN	0.3903	0.0350	0.1706	0.3917	0.0352	0.1714	0.4067	0.0371	0.1792
3-L-M NN	0.0628	0.0046	0.0286	0.0630	0.0046	0.0286	0.0671	0.0046	0.0293
3-L-L NN	0.0076	0.0004	0.0020	0.0076	0.0004	0.0020	0.0080	0.0004	0.0021
$\alpha = 0.01$									
3-L-S NN	0.0570	0.0066	0.0209	0.0572	0.0066	0.0210	0.0592	0.0069	0.0217
3-L-M NN	0.0271	0.0020	0.0080	0.0271	0.0021	0.0081	0.0277	0.0022	0.0083
3-L-L NN	0.0035	0.0003	0.0008	0.0035	0.0003	0.0008	0.0035	0.0003	0.0008

Variant level of noise in g with respect to different α and coarser to finer neural networks

Table 4.5: Optimal control on learning-informed PDEs with networks by 3 layers networks, but different sizes on the neurons (DoF), and a variant amount of training data.

4.7   Numerical results on optimal control of stationary Allen-Cahn equation

Next we study the optimal control of the Allen-Cahn equation, which involves a nonmonotone f and reads

$$-\Delta y + \frac{1}{\eta}(y^3 - y) = u \quad \text{in } \Omega, \quad \partial_\nu y = 0 \quad \text{on } \partial\Omega, \tag{4.65}$$

with $\eta > 0$. In our numerical tests, we set $\eta = 0.004$, use $\Omega = (0,2)^2$, and $h := 2^{-7}$.

We focus on 3-hidden-layer neural networks with 10, 12 and 10 neurons per layer yielding DoF= 293. In each hidden layer we use log-sigmoid transfer functions. Note also that since the input data here does not depend explicitly on the spatial variable x, i.e., $f = f(y)$, both the input and output layers have only one neuron, respectively. This is different to the previous test examples.

In our tests, we obtained the training data by solving the PDE in (4.65) with

$$u = u^d := \begin{cases} 1000, & x \in \Omega^l := (0,2) \times (0,1), \\ -1000, & x \in \Omega/\Omega^l. \end{cases}$$

In order to train the neural networks described above, the solution of the PDE is subsampled uniformly at a rate of 0.25, that is $H = 0.08$. As f has an one dimensional image space, it suffices that the data u^d correspond to a PDE solution that has a relatively wide range of values. Indeed, using our choice of u^d, the value of the corresponding solution y varies between -2.5 and 2.5 which turns out to be sufficient for learning f.

In Figure 1, we provide the plots of $F(z) = \int_{-1}^{z} f(t)\,dt$, the function f and its derivative f' on $[-K, K] \subset \mathbb{R}$, ($K = 10$ and $K = 2$, respectively) as well as their learned counterparts.

We observe that all the learning-informed versions preserve the key features of their exact counterparts very well. This is due to the fact that the training data cover exactly those ranges where important features are located.

As a next step, we consider the corresponding optimal control problem when the function f is replaced by its learned version. Notice that both the original and the learning-informed PDE admit no unique solution. Therefore the initial guess for the Newton iteration is crucial for the convergence to the final solutions. The algorithm for solving the optimal control problem for both PDEs is a combination of the semi-smooth Newton algorithm for (4.57) (with 0 as the initial guess) and the SQP algorithm. The switch between the solvers operates as follows: Consider the summed up residual of the four equations in (4.57) with respect to their norms in the spaces $H^{-1}(\Omega)$, $H^{-1}(\Omega)$, $L^2(\Omega)$ and $L^2(\Omega)$, respectively. Then we start our algorithm by calling the semi-smooth Newton iterations, and when the residual drops below a threshold value (e.g., 5 in our tests), then we switch to the SQP algorithm. The iteration is stopped if the residual is smaller than 10^{-10}, or a maximum of 30 iterations is reached. We fix $\alpha = 10^{-5}$ and
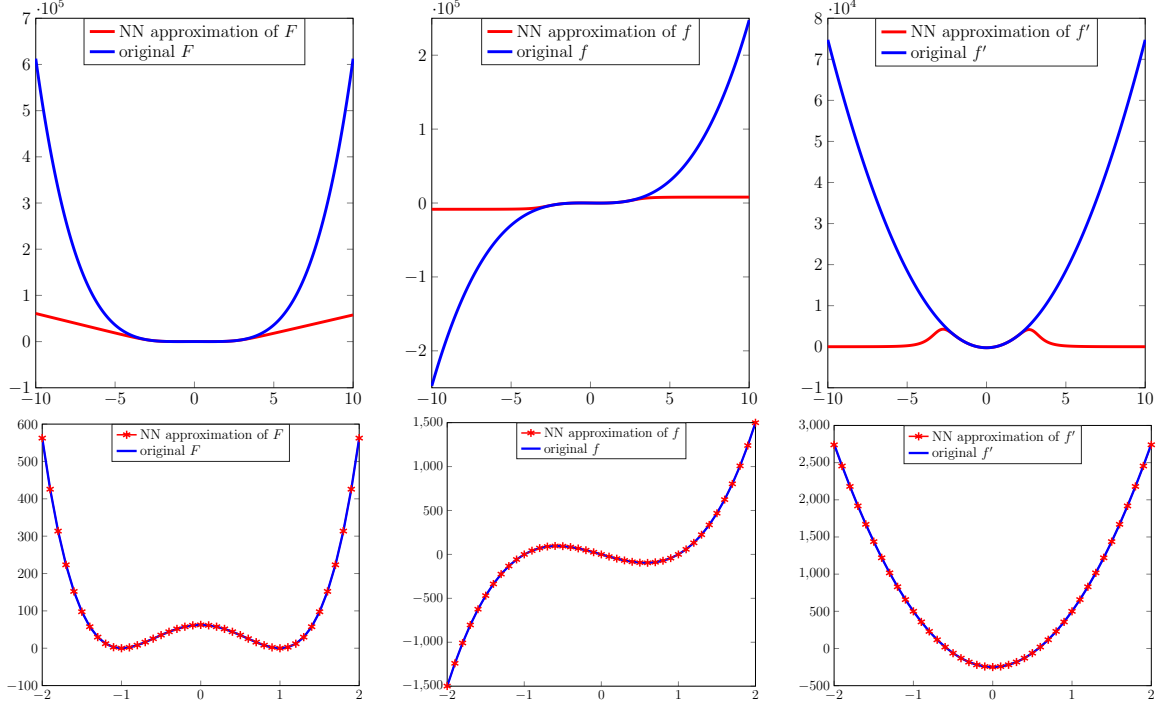
Figure 1: Functions F, f and its first order derivative f' along with the corresponding approximations learned from a neural network. We note that the range of the learning-informed function is influenced by the training data. The second row of images shows that the functions are well-approximated by their neural network counterparts in the ranges where the training data cover well, which here is around the interval $[-2, 2]$.

$\mathcal{C}_{ad} := \{u : -50 \leq u \leq 50\}$. Next consider g to be some polarized data preferring the values -1 and 1 and representing two distinct material states, e.g., a binary alloy; see Figure 3.
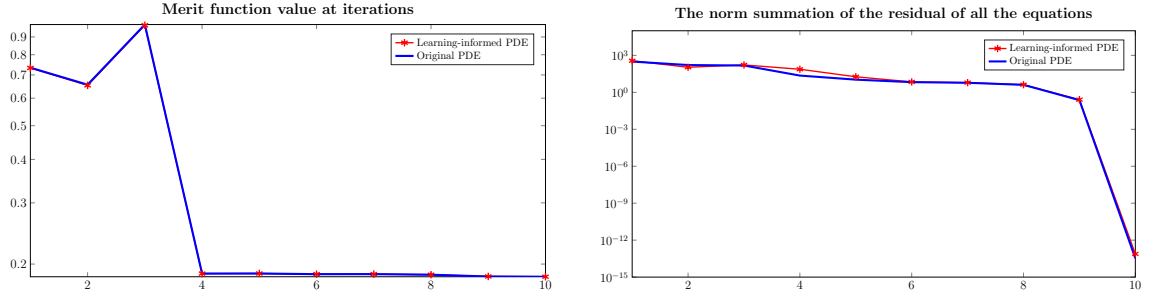
Figure 2: Merit function (left) and residual norm (right).

In Figure 2 we show the plots of the merit function values and also the overall residual of the first-order system in (4.57). The increasing part in the first few steps in the left plot (merit function) is due to the initilization of SSN while full step length is accepted. We notice that the threshold is reached by 10 overall iterations including also the SSN initialization steps.

Since neither the optimal control problem nor the PDE admit unique solutions, many local minima make the semi-smooth Newton algorithm rather sensitive to the initial guess. Concerning SQP we note here that enforcing the PDE and the box constraints too strongly in the early iterations, might result to the SQP algorithm getting trapped at some unfavorable stationary point. This has been numerically observed, e.g., when initializing the SQP algorithm by zero. In our tests, the combination of the semi-smooth Newton algorithm with the SQP algorithm, however, turns out to be robust against the aforementioned adverse effects. From Figure 3 (right plot) we observe a high accuracy approximation of the solutions of the learning-informed control to the solutions of the original control problem. Both, the PDE constraint
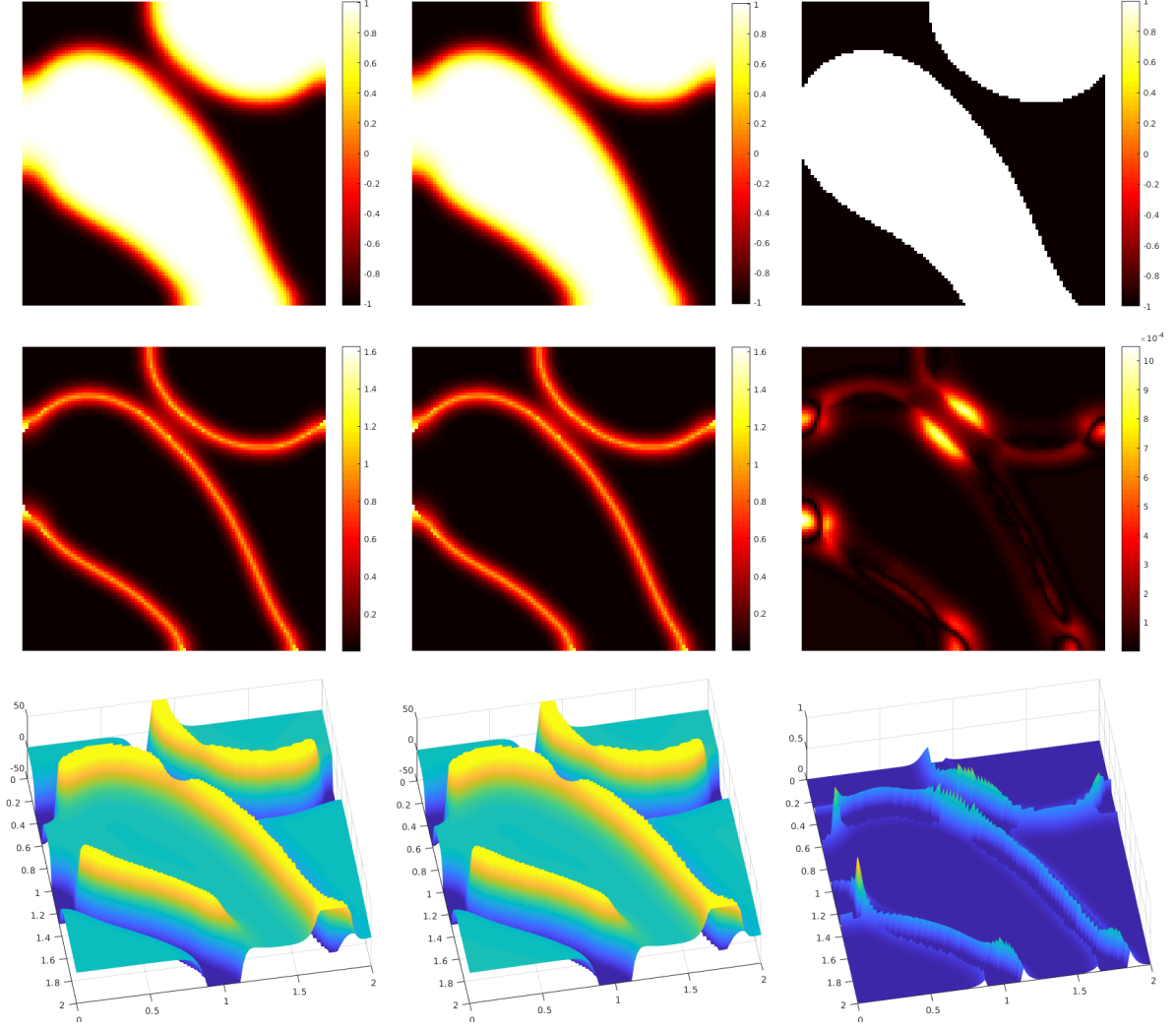
Figure 3: Optimal control of the stationary Allen-Cahn equation. First row: states (right: target data g; left and the middle: optimal states of learning-informed and original PDE, respectively); second row: difference images of states (left and the middle: differences (in absolute values) of optimal states to target state g; right: actual difference between the two optimal states $|y_\mathcal{N} - \bar{y}|$ in the first row; third row: left and middle the optimal controls corresponding to the learning-informed and original PDE respectively, as well as their difference $|u_\mathcal{N} - \bar{u}|$ on the right.

and also the box constraint are satisfied with high accuracy.

5 Application: Quantitative magnetic resonance imaging (qMRI)

According to [16], we consider the following optimization task in qMRI:

$$\text{minimize} \quad \frac{1}{2} \left\| P\mathcal{F}(y) - g^\delta \right\|_H^2 + \frac{\alpha}{2} \left\| u \right\|_U^2, \quad \text{over } (y, u := (T_1, T_2, \rho)^\top) \in Y \times U,$$

$$\text{s.t.} \quad \frac{\partial y}{\partial t}(t) = y(t) \times \gamma B(t) - \left(\frac{y_1(t)}{T_2}, \frac{y_2(t)}{T_2}, \frac{y_3(t) - \rho m_e}{T_1} \right), \quad t = t_1, \ldots, t_L, \qquad (5.1)$$

$$y(0) = \rho m_0,$$

$$u \in \mathcal{C}_{ad}.$$

where $0 < t_1 < \ldots < t_L$, $L \in \mathbb{N}$, $u \in U := [H^1(\Omega)]^3$ and $Y := [L^2(\Omega)^3]^L$ with $\Omega \subset \mathbb{R}^2$ the image domain, $H = \left[L^2(\mathbb{K})^2 \right]^L$ with $\mathbb{K}$ the Fourier space. By $\mathcal{F} : Y \to H$ we denote the component-wise

Fourier transform acting on (y_1, y_2), i.e., the first two coordinates of y, and $P : H \to H$ is a subsampling operator.

Further, $g^\delta = (g_l^\delta)_{l=1}^L \in H$ are (noisy) data, and $\mathcal{C}_{ad}$ is an nonempty, closed, convex, and bounded subset of $[L_\epsilon^\infty(\Omega)^+]^3$ with $L_\epsilon^\infty(\Omega)^+ := \{f \in L^\infty(\Omega) : \text{ess inf} f > \epsilon\}$, for some $\epsilon > 0$, which takes care of practical properties of physical quantities. The system of ordinary differential equations in (5.1) with initial value ρm_0 represents the renowned Bloch equations (BE), which model the evolution of nuclear magnetization in MRI [9] with the parameters γ and m_e being fixed constants. In our context, the external magnetic field B is assumed to be a uniformly bounded function in time. To accommodate different scaling, we consider $\frac{\alpha}{2} \|u\|_U^2 := \frac{\alpha_0}{2} \|u\|_{[L^2(\Omega)]^3}^2 + \frac{1}{2} |u|_{[H^1(\Omega)]^3}^2$, and

$$|u|_{[H^1(\Omega)]^3}^2 := \int_\Omega \left(\alpha_{1,1} |\nabla T_1|^2 + \alpha_{1,2} |\nabla T_2|^2 + \alpha_{1,3} |\nabla \rho|^2 \right) dx,$$

with $\alpha_0 > 0$ and $\alpha_{1,j} > 0$ for $j = 1, 2, 3$. For the ease of presentation, below we omit these scaling parameters.

Remark 5.1. *One readily checks that the solutions to the BE are bounded uniformly as long as T_1, T_2 are positive values and the magnetic field $B(t)$ is bounded. This property persists if either of the two terms on the right hand side of the equation is missing.*

Fixing the external magnetic field B according to an excitation protocol with a specific sequence of frequency pulses (cf., e.g., [16]) and associated echo times $\{t_i\}_{i=1}^L$ we have $u \mapsto \{y(t_i)\}_{i=1}^L$ yielding the solution map $\Pi : \mathcal{C}_{ad} \to [(L^\infty(\Omega))^3]^L$. Using this notation we have $Q(\cdot) = P\mathcal{F}(\Pi(\cdot))$. Noting that $\Pi(T_1, T_2, \rho) = \rho \Pi(T_1, T_2, 1)$ we show first continuity and differentiability results for $\tilde\Pi(\theta) := \Pi(T_1, T_2, 1)$ where $\theta := (T_1, T_2)^\top$. Even though for simplicity we do that for $\theta \in [L_\epsilon^\infty(\Omega)^+]^2$, with $\epsilon > 0$, we note that the map $\tilde\Pi$ can be continuously extended also for $T_1 = 0$ and/or $T_2 = 0$.

Proposition 5.2. *The operator $\tilde\Pi : [L_\epsilon^\infty(\Omega)^+]^2 \to [(L^\infty(\Omega))^3]^L$ is locally Lipschitz continuous, and Fréchet differentiable with locally Lipschitz derivative.*

Proof. Let $\theta, \theta^a \in [L_\epsilon^\infty(\Omega)^+]^2$ be given with associated solutions y, y^a of the BE, respectively. Suppressing $x \in \Omega$ in our notation, subtracting the BE for both θ values, and letting $r^a := y - y^a$ as well as $R(\theta) := \text{diag}(\frac{1}{T_2}, \frac{1}{T_2}, \frac{1}{T_1})$, we get

$$\frac{\partial r^a}{\partial t}(t) - r^a(t) \times \gamma B(t) + R(\theta)r^a = (R(\theta^a) - R(\theta))(y^a(t) - (0, 0, y_e))^\top, \quad r^a(0) = 0. \tag{5.2}$$

This equation and its homogeneous counterpart (i.e., with zero right hand side) admit unique solutions, respectively, cf. [43], for instance. According to [43, Theorem 3.12] the solution to (5.2) is

$$r^a(t) = \int_0^t \Phi(t, s)(R(\theta^a) - R(\theta))(y^a(s) - (0, 0, y_e)^\top)ds, \tag{5.3}$$

where $\Phi(t, s)$ is the principal matrix consisting of the three independent solutions of the homogeneous counterpart of (5.2) resulting from the initial data $h(s) = e_i$, $i = 1, 2, 3$, with $\{e_1, e_2, e_3\}$ the canonical orthonormal basis in $\mathbb{R}^3$. Note that it is easy to check that any such solution is uniformly bounded both in $t \geq 0$ and $\theta \geq 0$ almost everywhere. Since $R(\cdot)$ restricted to $[\epsilon, \infty)$ is Lipschitz (modulus $L > 0$), (5.3) can be further estimated as follows

$$|r^a(t)| \leq L \int_0^t |\Phi(t, s)(y^a(s) - (0, 0, y_e)^\top)|ds \, |\theta^a - \theta| \leq \tilde L(t) \, |\theta^a - \theta|,$$

for all $\theta^a, \theta \in [L_\epsilon^\infty(\Omega)^+]^2$. Note that the above estimate and in particular $\tilde L(t)$ can be considered independent of the spatial variable x due to the uniform bound on the solution of BE for every element of $\mathcal{C}_{ad}$ (cf. Remark 5.1). Therefore we have for some $L_\Pi > 0$ that

$$\|y^a(\cdot, t) - y(\cdot, t)\|_{[L^q(\Omega)]^3} \leq L_\Pi \|\theta^a - \theta\|_{[L^q(\Omega)]^2} \quad \text{for all } 1 \leq q \leq \infty.$$

By considering the above estimate at $\{t_i\}_{i=1}^L$ we get the asserted local Lipschitz continuity of $\tilde\Pi$.

We now proceed to Fréchet differentiability. Let $\theta \in [L_\epsilon^\infty(\Omega)^+]^2$, $v \in [L^\infty(\Omega)]^2$ be an arbitrary vector, and let $\theta^a = \theta + av$ where $a > 0$ is such that $\theta^a \in [L_\epsilon^\infty(\Omega)^+]^2$. Dividing (5.2) by a and letting $p_\theta^a := \frac{r^a}{a}$, we get:

$$\frac{\partial p_\theta^a}{\partial t}(t) - p_\theta^a(t) \times \gamma B(t) + R(\theta)p_\theta^a = \frac{(R(\theta^a) - R(\theta))}{a}(y^a(t) - (0,0,y_e))^\top, \quad p_\theta^a(0) = 0. \qquad (5.4)$$

Existence, uniqueness and representation of a solution again follows from [43, Theorem 3.12]:

$$p_\theta^a(t) = \int_0^t \Phi(t,s)\frac{(R(\theta + av) - R(\theta))}{a}(y^a(s) - (0,0,y_e)^\top)ds.$$

Recall that $R(\cdot)$ is continuously differentiable for $\theta > 0$ and time independent. For $a \downarrow 0$ and $p_\theta := \lim_{a\to 0} p_\theta^a$, we have

$$p_\theta(t) = \int_0^t \Phi(t,s)R'(\theta;v)(y(s) - (0,0,y_e)^\top)ds,$$

where $R'(\theta;v)$ denotes the directional derivative of R at θ in direction v. By considering again the uniform boundedness with respect to the spatial variable and pointwise evaluation at $\{t_i\}_{i=1}^L$, we get that $p_\theta = \tilde{\Pi}'(\theta;v)$ is bounded, and also linear with respect to the direction $v \in [L^\infty(\Omega)]^2$. Thus, $\tilde{\Pi}$ is Gateaux differentiable. Notice further that, due to $R'(\cdot;v)$ being locally Lipschitz, we have also the local Lipschitz continuity (modulus $L_{p_\theta} > 0$) of the directional derivative:

$$|p_{\theta^a} - p_\theta|^q \le L_{p_\theta}^q |\theta^a - \theta|^q \|v\|_{[L^\infty(\Omega)]^2} \quad \text{for all } \theta^a, \theta \in [L_\epsilon^\infty(\Omega)^+]^2, \text{ and } 1 \le q \le \infty, \qquad (5.5)$$

with the above estimate again independent of the spatial variable. This together with the linearity of the Gateaux derivative implies the Fréchet differentiability of $\tilde{\Pi}$. Finally we also conclude the Lipschitz continuity of the Fréchet derivative:

$$\left\|(\tilde{\Pi}'(\theta^a) - \tilde{\Pi}'(\theta))v\right\|_{[L^\infty(\Omega)]^{3L}} \le L_{p_\theta} \|\theta^a - \theta\|_{[L^\infty(\Omega)]^2} \|v\|_{[L^\infty(\Omega)]^2}. \qquad (5.6)$$

This ends the proof. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Note that the continuity and differentiability of $\Pi = \rho\tilde{\Pi}$ for $u \in \mathcal{C}_{ad}$ follows readily as $\rho \in L^\infty(\Omega)$. As a consequence, existence of a solution to (5.1) can be shown similarly to Proposition 2.1.

Remark 5.3. *The estimate (5.5) indicates that for every $u = (\theta^\top, \rho)^\top \in \mathcal{C}_{ad}$, and $h \in [L^\infty(\Omega)]^2$ sufficiently small, we even have*

$$\left\|\tilde{\Pi}(\theta + h) - \tilde{\Pi}(\theta) - \tilde{\Pi}'(\theta)h\right\|_{[L^q(\Omega)]^{3L}} = \mathcal{O}(\|h\|_{[L^q(\Omega)]^2}^2) \quad \text{for all } 1 \le q \le \infty.$$

We also note that due to properties of the Bloch operator, we have that both $\tilde{\Pi}'(\theta) : [L^2(\Omega)]^2 \to [L^2(\Omega)]^{3L}$ and $Q'(u) : [L^2(\Omega)]^3 \to [(L^2(\mathbb{K}))^2]^L$ are bounded linear operators, respectively, as soon as $u = (\theta^\top, \rho)^\top \in \mathcal{C}_{ad}$. In this sense, we consider in the following $\tilde{\Pi}'(\theta)$ and $Q'(u)$ to be elements in $\mathcal{L}([L^2(\Omega)]^2, Y)$ and $\mathcal{L}(U, H)$, respectively.

We are now interested in finding a data-driven approximation $\Pi_{\mathcal{N}}(u) := \rho\mathcal{N}(T_1, T_2)$ of Π and in solving the reduced problem

$$\begin{aligned}
\text{minimize} \quad &\frac{1}{2}\left\|Q_{\mathcal{N}}(u) - g^\delta\right\|_H^2 + \frac{\alpha}{2}\|u\|_U^2, \quad \text{over } u \in U, \\
\text{s.t.} \quad &u = (T_1, T_2, \rho)^\top \in \mathcal{C}_{ad},
\end{aligned} \qquad (5.7)$$

with $Q_{\mathcal{N}}(u) = P\mathcal{F}(\Pi_{\mathcal{N}}(T_1, T_2, \rho))$. Existence of a solution to (5.7) can again be argued similarly to Proposition 2.1.

We finish this section with the corresponding approximation result.

Proposition 5.4. *Let $\theta = (T_1, T_2)^\top$, $u = (\theta^\top, \rho)^\top \in \mathcal{C}_{ad}$. Assume the following error bounds in the neural network approximations*

$$\left\|\mathcal{N}(\theta) - \tilde{\Pi}(\theta)\right\|_{[L^\infty(\Omega)^3]^L} \le \epsilon \quad \text{and} \quad \left\|\mathcal{N}'(\theta) - \tilde{\Pi}'(\theta)\right\|_{\mathcal{L}([L^2(\Omega)]^2, [L^\infty(\Omega)^3]^L)} \le \epsilon_1,$$

Then we have

$$\|Q(u) - Q_{\mathcal{N}}(u)\|_H \le C\epsilon, \tag{5.8}$$

$$\|Q'(u) - Q'_{\mathcal{N}}(u)\|_{\mathcal{L}(U,H)} \le C_1\epsilon + C_2\epsilon_1, \tag{5.9}$$

for some positive constants C, C_1 and C_2 which are all independent of ϵ and ϵ_1.

Before we commence with the proof, note that the above assumptions are plausible in view of $u \in \mathcal{C}_{ad} \subset [(L_\epsilon^\infty(\Omega))^+]^3$ and Theorems 3.1 and 3.2.

Proof. The first estimate is straightforward from the definition of Q

$$\|Q(u) - Q_{\mathcal{N}}(u)\|_H = \left\| P\mathcal{F}(\rho(\mathcal{N}(\theta) - \tilde{\Pi}(\theta))) \right\|_H \le \left\| \rho(\mathcal{N}(\theta) - \tilde{\Pi}(\theta)) \right\|_{[L^2(\Omega)^3]^L} \le C\epsilon, \tag{5.10}$$

since $\mathcal{C}_{ad} \subset [L^\infty(\Omega)]^3$ is a bounded set.

To see the second estimate, notice that for every $v := (v_1, v_2, v_3)^\top \in [L^2(\Omega)]^3$,

$$Q'(u)v = P\mathcal{F}(v_1 \tilde{\Pi}(\theta)) + P\mathcal{F}(\rho\tilde{\Pi}'(\theta)(v_2, v_3)^\top), \tag{5.11}$$

and similarly for $Q'_{\mathcal{N}}$. Thus,

$$
\begin{aligned}
\|(Q'(u) - Q'_{\mathcal{N}}(u))v\|_H \le{} & C_1 \left\| \mathcal{N}(\theta) - \tilde{\Pi}(\theta) \right\|_{[L^\infty(\Omega)^3]^L} \|v_1\|_{L^2(\Omega)} \\
& + C_2 \left\| \mathcal{N}'(\theta) - \tilde{\Pi}'(\theta) \right\|_{\mathcal{L}([L^2(\Omega)]^2, [L^\infty(\Omega)^3]^L)} \|(v_2, v_3)\|_{[L^2(\Omega)]^2},
\end{aligned}
$$

which ends the proof. $\qquad\square$

Finally, we show the Lipschitz continuity of Q and Q'. For the learning-informed versions this is done similarly. Using the isometric property of the Fourier transform and the triangle inequality, we get for every $u_a, u_b \in \mathcal{C}_{ad}$ and some $C \ge 1$:

$$\|Q(u_a) - Q(u_b)\|_H \le C \left(\|\rho_a - \rho_b\|_{L^2(\Omega)} + \|\theta_a - \theta_b\|_{[L^2(\Omega)]^2} \right).$$

Similarly, we estimate $\|(Q'(u_a) - Q'(u_b))v\|_H$ assuming that v is unitary:

$$
\begin{aligned}
& \|(Q'(u_a) - Q'(u_b))v\|_H \\
\le{} & \left\| P\mathcal{F}(v_1(\tilde{\Pi}(\theta_1) - \tilde{\Pi}(\theta_2))) \right\|_H + \left\| P\mathcal{F}\left((\rho_1\tilde{\Pi}'(\theta_1) - \rho_2\tilde{\Pi}'(\theta_2))[v_2, v_3] \right) \right\|_H \\
\le{} & L_{\tilde{\Pi}} \|\theta_1 - \theta_2\|_{[L^2(\Omega)]^2} + \|\rho_1 - \rho_2\|_{L^\infty(\Omega)} + L_{p_\theta} \|\rho_2\|_{L^\infty(\Omega)} \|\theta_1 - \theta_2\|_{[L^2(\Omega)]^2}.
\end{aligned}
$$

Here, we use the fact that $\mathcal{F}$ is a unitary operator, $\|\tilde{\Pi}(\theta)\|_{[L^\infty(\Omega)^3]^L}$ is uniformly bounded, and $L_{\tilde{\Pi}}$ and L_{p_θ} are the Lipschitz constants of $\tilde{\Pi}(\theta)$ and $\tilde{\Pi}'(\theta)$, respectively.

5.1 Numerical algorithm

For the numerical solution of the reduced optimization problem associated with the present qMRI problem, we adopt the SQP method, i.e., Algorithm 1, from the previous application to the qMRI setting. The only difference is that we do not need the Newton iterations in Step $(a1)$ there. Recall that now we have $u = (T_1, T_2, \rho)^\top$. In comparison to the previous PDE examples, the sensitivity of the reduced objective functional in (5.7) is directly available as

$$\mathcal{J}'_{\mathcal{N}}(u) = (\rho(\mathcal{N}'(T_1, T_2))^*, \mathcal{N}(T_1, T_2))^\top \mathcal{F}^*(\mathcal{F}(\rho\mathcal{N}(T_1, T_2)) - g) + \alpha(\mathrm{Id} - \Delta)(T_1, T_2, \rho)^\top. \tag{5.12}$$

Further, in every QP-step one is confronted with solving

$$
\begin{aligned}
\text{minimize} \quad & \langle \mathcal{J}'_{\mathcal{N}}(u_k), h \rangle_{U^*, U} + \frac{1}{2} \langle H_k(u_k)h, h \rangle_{U^*, U} \quad \text{over } h \in U \\
\text{s.t.} \quad & u_k + h \in \mathcal{C}_{ad},
\end{aligned}
\tag{5.13}
$$

where now $H_k(u_k)$ is the following symmetrized version of the Hessian of $\mathcal{J}_\mathcal{N}$ at $u_k \in \mathcal{C}_{ad}$:

$$(\rho(\mathcal{N}'(T_1, T_2))^*, \mathcal{N}(T_1, T_2))^\top \mathcal{F}^* \mathcal{F}(\rho(\mathcal{N}'(T_1, T_2)), \mathcal{N}(T_1, T_2)) + \alpha(\mathrm{Id} - \Delta).$$

In the following tests, we choose $\mu_0 = 1$, $\epsilon = 10^{-5}$, $r = 0.618$, $\kappa = 10^{-3}$, and $\xi = 0.5$. We stop the SQP iteration when the norm of the residuals of the first-order optimality system drops below a user-specified threshold value of 10^{-3} or a maximum of 40 iterations is reached. The regularization parameter is $\alpha_0 = [1, 1, 1] \times 10^{-10}$ for the L^2 part in the regularization functional in (5.7), and $\alpha_1 = [1, 20, 2] \times 10^{-9}$ for the H^1 seminorm part in (5.7), with respect to T_1, T_2, ρ, respectively. The parameter c in the complementary constraint is chosen to be $10^9 \alpha_1$ in the numerical tests, which is different to the previous examples. The values of all remaining parameters in Algorithm 1 not explicitly mentioned here, are kept the same as in the previous tests. We notice here that due to the analytical structure of the problem, the primal-dual active set algorithm for this example is equivalent to a SSN approach only in the discretized setting. We refer to [26] for a path-following SSN solver which works in function space upon Moreau-Yosida regularization of the indicator function of the constraint set.

5.2 Numerical results on qMRI

For the generation of the training data, we use the explicit Bloch dynamics of [14] where a specific pulse sequence with acronym IR-bSSFP (short for *Inversion Recovery balanced Steady State Free Precession*) is considered. Let $(M_l)_{l=1}^L$ denote the pertinent explicit solution. This yields $\Pi(u) = \rho(M_l(T_1, T_2))_{l=1}^L$, with $u = (T_1, T_2, \rho)^\top$. The MRI tests are implemented based on an anatomical brain phantom, publicly available from the Brain Web Simulated Brain Database [1, 12]. We use a slice with 217×181 pixels from this database and cut some of the zero fill-in pixels so that we finally arrive at a 181×181-pixel image. The selected range for u reflects natural values encountered in the human body. This gives rise to the box constraint $\mathcal{C}_{ad} := \{u = (T_1, T_2, \rho)^\top : T_1 \in (0, 5000), T_2 \in (0, 1800), \rho \in (0, 6000)\}$. In Figure 4, we show the images from the brain phantom for ideal parameter maps T_1, T_2 and ρ.
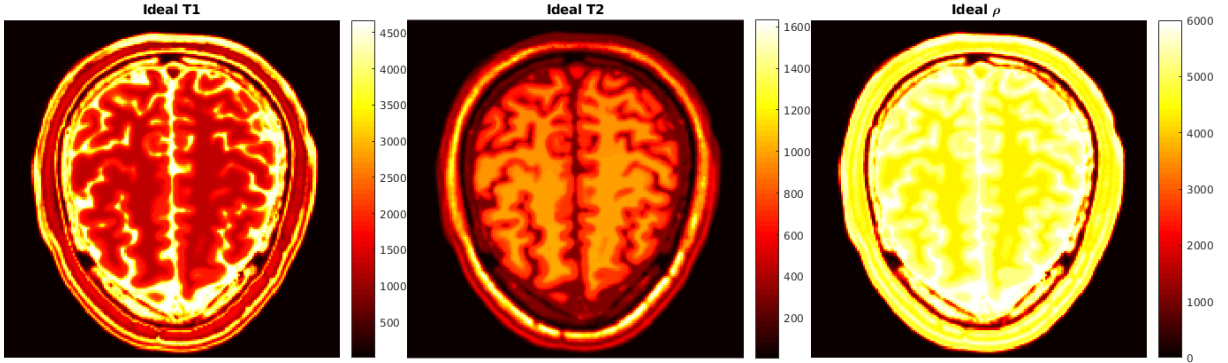

Figure 4: Simulated ideal tissue parameters of a brain phantom.

Loss function and training method   For each residual of two neighbored images in the time series, we use the mean squared error as the loss function and the Bayesian regularization algorithm based on the Levenberg-Marquardt method for the training of the residual neural networks DRNN described below. The learning algorithm and the setting are the same as the previous examples.

Architecture of the network   In order to approximate the Bloch solution map, we use Direct Residual Neural Networks (DRNNs). Here the solution map at a given time is approximated by a neural network depending only on the initial condition M_0. To explain this in detail, let $\hat{M}$ be the learned approximation of M, i.e. $\hat{M}_l(T_1, T_2) \simeq M_l(T_1, T_2)$, $l = 1, \ldots, L$. The DRNN framework then reads:

$$\hat{M}_l(T_1, T_2) = \hat{M}_0(T_1, T_2) + \mathcal{N}_{\Theta_l}(T_1, T_2), \quad l = 1, \ldots, L, \quad \hat{M}_0(T_1, T_2) = M_0, \tag{5.14}$$

with sub-networks $\{\mathcal{N}_{\Theta_l}\}_{l=1}^L$. The map $(M_l)_{l=1}^L$ is then simply approximated by the map $(M_0 + \mathcal{N}_{\Theta_l})_{l=1}^L$.

We use sub-networks with a total number of hidden layers equal to 1, 2, or 3. In each case, we design the architecture at every layer so that the total degrees of freedom in Θ are essentially the same. The detailed description is summarized in Table 5.1. In total, we test 9 different architectures. For every network, we use the 'softmax' activation function in the layer next to the output layer, and the 'logsigmoid' function in all other hidden layers. The difference to the previous optimal control examples is that the architecture applies to every sub-network which is of residual type, as described above.

	HL 1	HL 2	HL 3	DoF	HL 1	HL 2	HL 3	DoF	HL 1	HL 2	HL 3	DoF
	Small DoF				Medium DoF				Large DoF			
1-L-NN	24	-	-	122	75	-	-	377	130	-	-	652
2-L-NN	7	10	-	123	17	16	-	373	23	22	-	643
3-L-NN	5	8	5	120	10	15	10	377	15	18	15	650

Table 5.1: The architecture of every sub-network. Both input and output layers have two neurons.

Training and validation data The training including also the validation data are generated from the dictionary which has been used in methods for magnetic resonance fingerprinting (MRF), e.g., [14, 32]. These are time series resulting from the dynamics, such as e.g. IR-bSSFP, which was introduced in [41], given the initial value $M_0 = (0, 0, -1)$. We fix the length of the pulse sequence to be $L = 20$. Of course, other numerical simulations of the Bloch equations can also be proper options as input-output training data. We test each of the networks with architectures according to Table 5.1 using three levels of training data, which we term 'small', 'medium' and 'large'. For the small size training data, we generate parameter values for (T_1, T_2) from $D_1 := (0 : 400 : 5000)$ and $D_2 := (0 : 100 : 1800)$ (in MATLAB notation) which contribute 247 entries of time series; for the medium size training data from $D_1 := (0 : 200 : 5000)$ and $D_2 := (0 : 50 : 1800)$ with a total of 962 entries; and for the large size data $D_1 := (0 : 50 : 5000)$ and $D_2 := (0 : 20 : 1800)$ resulting in total in 9191 entries. The input data of the neural networks consist of elements of the set $D_1 \times D_2$. Note here that we include 0 for both T_1 and T_2, respectively, to take care of the marginal area in the imaging domain. The output data will be the Bloch dynamics corresponding to each pair of elements in $D_1 \times D_2$. Both input and output data are normalized to pairs whose elements take values in the range $[-1, 1]$. This is done by `mapminmax` function in MATLAB.

For the SQP we consider the image domain to be $[0, 1] \times [0, 1]$, thus the spatial discretization size is $h = 1/180$. We compare the results of the learning-based method with results from the algorithm proposed in our previous work [16]. The initialization to the SQP algorithm and also the algorithm in [16] is done by using the so-called BLIP algorithm of [14] with a dictionary resulting from the small size $D_1 \times D_2$. The parameters are tuned as in [16]. Concerning the degradation of our image data we consider here Gaussian noise of mean 0 and standard deviation 30.

Concerning the results reported in Table 5.2, the columns of $M(\theta)$ reflect the approximation accuracy to the discrete dynamical Bloch sequences using various neural networks. A smaller value refers to a smaller error, or in other words to higher accuracy in the approximation. However, higher accuracy in the Bloch solution operator approximation does not necessarily result in a better estimation of the T_1, T_2 parameters. For this purpose, note that differently to the previous example, here the error is evaluated against the ideal solutions. The dashes in Table 5.2 belong to cases where the training data are not sufficient to guarantee well enough learning under the current setting our paper. We observe that the results are varying slightly under different network architectures and also when using different volumes of training data. In particular, we have the observations: (i) When the training data is sufficiently rich, with the same number of hidden layers, then the larger the number of neurons the better becomes the approximation to the Bloch mapping. However, this does not mean necessarily better to the estimated parameters in terms of the error rates provided. (ii) We find that the small DoF networks with small volume training data achieve already almost the same accuracy as the ones using medium and large DoF networks. The results are almost as good as using SQP with the exact Bloch solution formula. We have also observed that the SQP method with learning-based operators can be computationally more efficient than the one with the exact Bloch operators. This is due to the fact that evaluating the learning-based operator can be much cheaper than solving the exact physical model, although a learning process has to be performed before-hand.

	Small DoF				Medium DoF				Large DoF			
	T_1	T_2	ρ	$M(\theta)$	T_1	T_2	ρ	$M(\theta)$	T_1	T_2	ρ	$M(\theta)$
	Small training data											
1 Layer NN	0.084	0.056	0.004	0.016	–	–	–	–	–	–	–	–
2 Layer NN	0.093	0.054	0.005	0.013	–	–	–	–	–	–	–	–
3 Layer NN	0.087	0.052	0.009	0.012	–	–	–	–	–	–	–	–
	Medium training data											
1 Layer NN	0.084	0.058	0.003	0.004	0.089	0.052	0.002	0.005	–	–	–	–
2 Layer NN	0.143	0.060	0.006	0.004	0.090	0.052	0.005	0.003	–	–	–	–
3 Layer NN	0.086	0.051	0.003	0.004	0.087	0.051	0.004	0.002	–	–	–	–
	Large training data											
1 Layer NN	0.120	0.078	0.005	0.002	0.120	0.081	0.004	0.0014	0.090	0.050	0.004	0.0009
2 Layer NN	0.094	0.057	0.006	0.001	0.094	0.043	0.002	0.002	0.089	0.056	0.004	0.0012
3 Layer NN	0.096	0.059	0.005	0.0007	0.087	0.051	0.004	0.0004	0.087	0.051	0.004	0.0006
Method [16]	0.102	0.094	0.004	–	proposed Algorithm using exact Bloch				0.084	0.051	0.003	–

For 25% Cartesian subsampled k-space data with Gaussian noise of mean 0 and standard deviation 30. Relative error computed from $\frac{\|x-x^*\|}{\|x^*\|}$ for $x = T_1,\ T_2,\ \rho,\ M$ where $\|\cdot\|$ is the discrete 2-norm.

Table 5.2: Error comparison for qMRI: Using Bloch maps by networks with different layers, different size of neurons, and a variant of training data

In Figures 5 and 6, we provide visual comparison of results from different methods for quantitative MRI. Particularly, we compare to the method proposed by the authors in [16] assuming knowledge of the exact Bloch solution map and also the BLIP algorithm in [14] in which the fine dictionary (i.e., a large size data set) is used.

The images produced by the proposed algorithm with a learning-informed model are based on the 1-hidden-layer network with a small size of DoF which is trained with medium volume data. We can see that the proposed approach clearly gives better results for the recovering of the quantitative parameters when compared with the methods in [16] and BLIP [14]. In particular, we observe the T_1, T_2 parameters estimated by the proposed method are significantly better than the results from the other two methods in terms of spatial regularity. In particular, some artifacts are avoided by the proposed method. This is due to using an H^1 term for u in the objective while the method in [16], for instance, uses an L^2 term only.

We notice that the method in [16] is superior only if the noise in the data is small. The learning-informed operator could also be applied yielding results similar to those of the original method [16]. Since for real MRI experiments, the k-space data may be contaminated by different sources of noise, certain spatial regularization could help to stabilize solutions. The proposed method in this paper seems to be new to qMRI in this respect, since previous methods typically use pixel-wise estimation so that spatial regularity is harder to enforce. Along this line, one may consider more sophisticated regularization methods such as, e.g., total variation or total generalized variation regularization, to take care of spatial discontinuities. Such a study, however, is clearly beyond the scope of the present paper.

6   Conclusion

In this paper, we have proposed and analyzed a general optimization scheme for solving optimal control problems subject to constraints which are governed by learning-informed differential equations. The applications and numerical tests have verified the feasibility of the proposed scheme for two key applications. We envisage that our work will provide a fundamental framework for dealing with physical models whose underlying differential equation is partially unknown and thus needed to be learned by data, with the latter typically obtained from experiments or measurements. Our approach avoids learning the full model, i.e., learning directly the solution of the overall minimization problem as this could be on the one hand too complicated and on the other, it could render the method more towards being a black box solver. By learning only a component, i.e., a nonlinearity, or the solution map of the underlying differential equation, the method is kept more faithful to the true physics-based model.

An important factor for the applicability of the proposed framework is the learnability of the operator resulting from differential equations. We observed that the uniform boundedness of the range of the
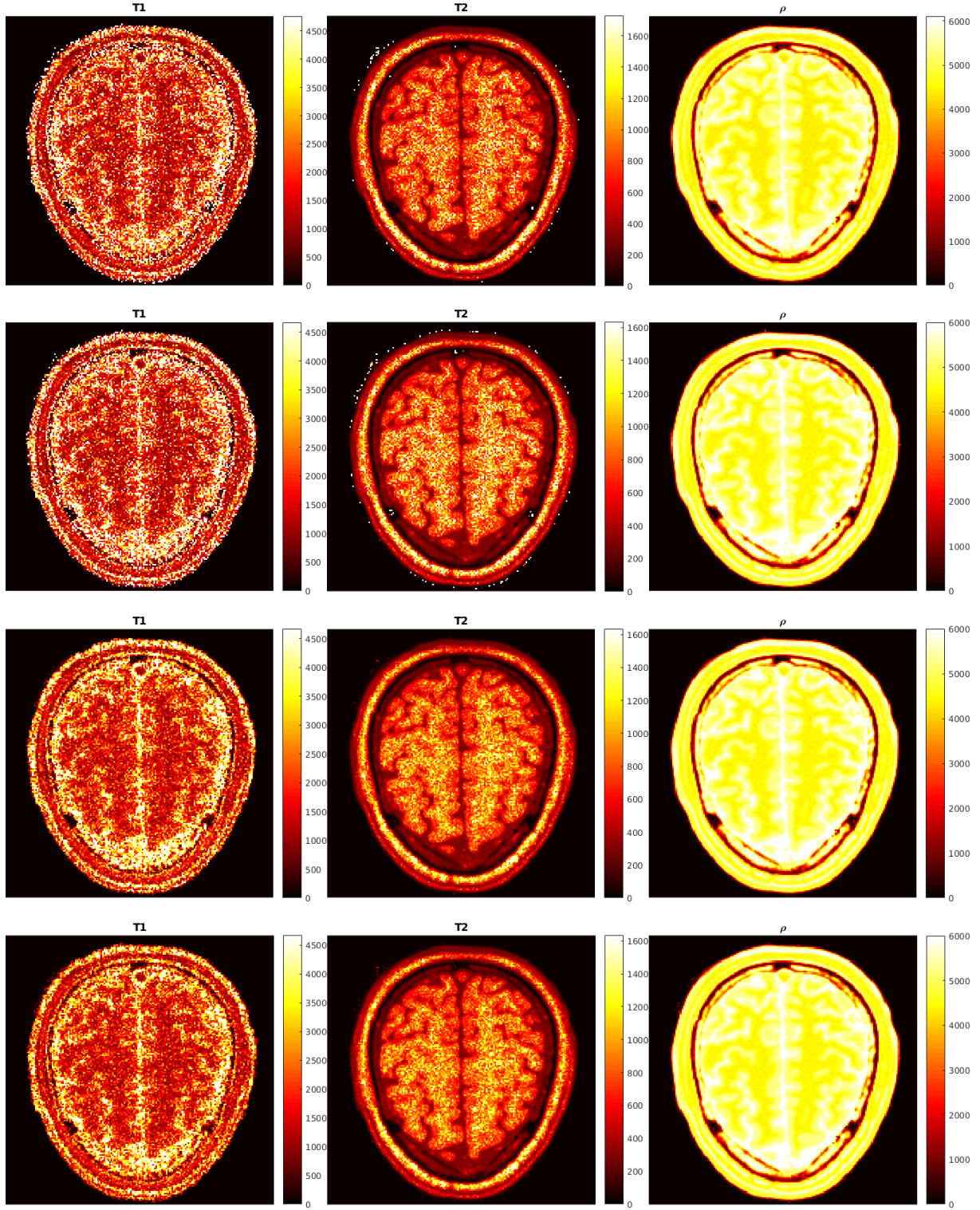
Figure 5: Estimated tissue parameters from subsampled and noisy measurements. First row: Solution using the BLIP method in [14] using a fine dictionary; Second row: Solution using method in [16]; Third row: Our SQP solution with learning-informed model small size DoF, 1-hidden-layer residual networks and trained with medium size data. Fourth row: Our SQP solution using the analytical formula for the Bloch solution map.

input and output data (state variable) played a crucial role, stemming from the fact that the density of
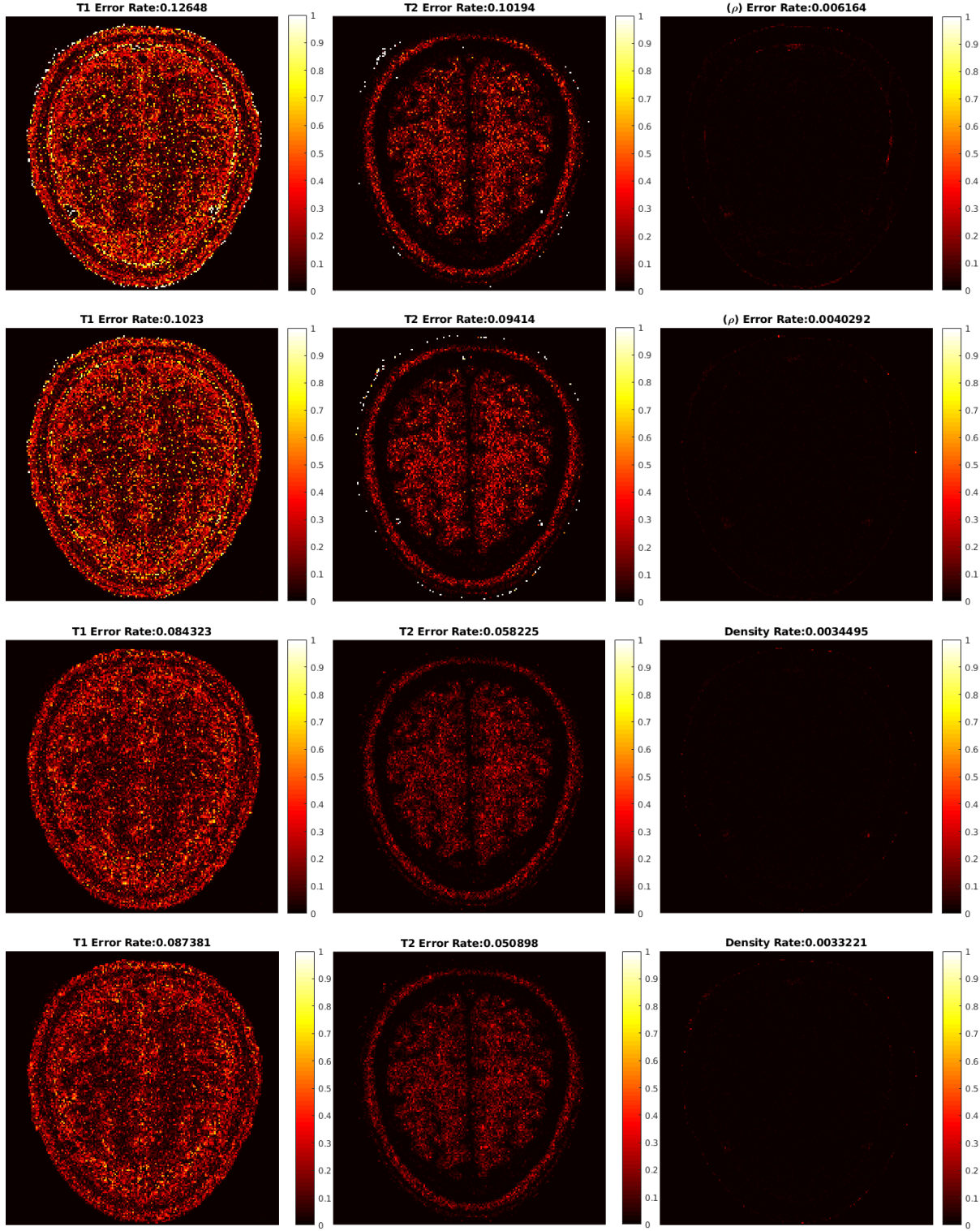
Figure 6: Relative errors of the estimated tissue parameters from subsampled and noisy measurements. First row: Error map from BLIP [14] using a fine dictionary; Second row: Error map from [16]; Third row: Error map for our SQP solution with learning-informed model. Fourth row: Error map for our SQP solution with exact formula for the Bloch map as [16]. All errors are normalized.

neural networks holds in the topology of uniform convergence on compact sets. As we observed in the double-well potential example, learning the nonlinearity in its whole range is not necessarily needed, but

only in a range in which the state variables lie, with this range being known due to a priori estimates. Indeed, in the stationary Allen-Cahn control problem, the learning is only performed over a very local part of the nonlinearity (the double-well part), giving an almost perfect result. This shows some potential for reducing the training load by properly analyzing the properties of the nonlinearities. From the quantitative MRI example we furthermore observed that the embedding of the learned operator in the reconstruction process led to a reduction in the computational load, since it avoids a repetitive solution of the exact physical model.

A series of future studies arise from the present work. The analysis implemented here asks for smooth neural networks approximating (part of) the control-to-state map. A theory incorporating nonsmooth neural networks is an important extension as this will include networks with ReLU activation functions. Further studies can also incorporate the network structure (in the spirit of optimal experimental design) as well as aspects of the training process into the overall minimization process to further optimize and robustify the new technique. Finally, the errors due to the early stopping of the numerical algorithm as well as due to the ones from the numerical discretization, can be incorporated in the a priori error analysis. This could be of benefit for designing more suitable network architectures.

Acknowledgments

References

[1] Brainweb: Simulated brain database. *http://www.bic.mni.mcgill.ca/brainweb/*.

[2] R.A. Adams and J.J.F. Fournier. *Sobolev spaces*, volume 140 of *Pure and Applied Mathematics (Amsterdam)*. Elsevier/Academic Press, Amsterdam, second edition, 2003.

[3] J. Adler and O. Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017. `https://doi.org/10.1088/1361-6420/aa9581`.

[4] C.D. Aliprantis and K.C. Border. *Infinite dimensional analysis, a Hitchhiker's Guide.* Springer, 2006.

[5] S. Arridge, P. Maass, O. Öktem, and C.B. Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28(1):1–174, 2019. `https://doi.org/10.1017/S0962492919000059`.

[6] H. Attouch, G. Buttazzo, and G. Michaille. *Variational analysis in Sobolev and BV spaces*, volume 17 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA, second edition, 2014.

[7] B. Baker, O. Gupta, N. Naik, and R. Raskar. Designing neural network architectures using reinforcement learning. *Conference paper on International Conference on Learning Representations*, pages 1–18, 2017. `https://openreview.net/pdf?id=S1c2cvqee`.

[8] F. Balsiger, A. Shridhar Konar, S. Chikop, V. Chandran, O. Scheidegger, S. Geethanath, and M. Reyes. Magnetic resonance fingerprinting reconstruction via spatiotemporal convolutional neural networks. In Rueckert D. Knoll F., Maier A., editor, *Machine Learning for Medical Image Reconstruction. MLMIR 2018*, volume 11074 of *LNCS*, pages 39–46. Springer, Cham, 2018. `https://doi.org/10.1007/978-3-030-00129-2_5`.

[9] F. Bloch. Nuclear induction. *Physical Review*, 70:460–473, 1946.

[10] L. Bottou, F.E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[11] A. Braides. Convergence of local minimizers. In *Local Minimization, Variational Evolution and Γ-Convergence*, pages 67–78. Springer, 2014.

[12] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, N.J. Kabani, C.J. Holmes, and A.C. Evans. Design and construction of a realistic digital brain phantom. *IEEE Transactions on Medical Imaging*, 17(3):463–468, 1998. https://doi.org/10.1109/42.712135.

[13] G. Dal Maso. *Introduction to Γ-convergence*. Birkhäuser, 1993.

[14] M. Davies, G. Puy, P. Vandergheynst, and Y. Wiaux. A compressed sensing framework for magnetic resonance fingerprinting. *SIAM Journal on Imaging Sciences*, 7(4):2623–2656, 2014. https://doi.org/10.1137/130947246.

[15] S.P. Dirkse and M.C. Ferris. The path solver: A non-monotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software*, 5(4):123–156, 1995. https://doi.org/10.1080/10556789508805606.

[16] G. Dong, M. Hintermüller, and K. Papafitsoros. Quantitative magnetic resonance imaging: From fingerprinting to integrated physics-based models. *SIAM Journal on Imaging Sciences*, 12(2), 2019. https://doi.org/10.1137/18M1222211.

[17] W. E. A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 5:1–11, 2017. https://doi.org/10.1007/s40304-017-0103-z.

[18] L.C. Evans. *Partial Differential Equations*, volume 19 of *Graduate studies in mathematics*. American Mathematical Society, second edition, 2010.

[19] H.O. Fattorini. *Infinite-dimensional optimization and control theory*, volume 62 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1999. https://doi.org/10.1017/CBO9780511574795.

[20] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[21] E. Haber and L. Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34:014004, 2018. https://doi.org/10.1088/1361-6420/aa9a90.

[22] J. Han, A. Jentzen, and W. E. Solving high-dimensional partial differential equations using deep learning. *Proceeding of National Academy of Science*, 115(34):8505–8510, 2018. https://doi.org/10.1073/pnas.1718942115.

[23] M. Hanke. The regularizing Levenberg-Marquardt scheme is of optimal order. *Journal of Integral Equations and Applications*, 22:259–283, 2010. https://doi.org/10.1216/JIE-2010-22-2-259.

[24] M. Hintermüller. Mesh independence and fast local convergence of a primal-dual active-set method for mixed control-state constrained elliptic control problems. *The ANZIAM Journal*, 49:1–38, 2007. https://doi.org/10.1017/S1446181100012657.

[25] M. Hintermüller, K. Ito, and K. Kunisch. The primal-dual active set strategy as a semismooth Newton method. *SIAM Journal on Optimization*, 13:865–888, 2002. https://doi.org/10.1137/S1052623401383558.

[26] M. Hintermüller and K. Kunisch. Feasible and noninterior path-following in constrained minimization with low multiplier regularity. *SIAM Journal on Control and Optimization*, 45(4):1198–1221, 2006. https://doi.org/10.1137/050637480.

[27] M. Hintermüller and M. Ulbrich. A mesh-independence result for semismooth Newton methods. *Mathematical Programming, Series B*, 101:151–184, 2004. https://doi.org/10.1007/s10107-004-0540-9.

[28] M. Leshno, V.Y. Lin, A. Pinkus, and S. Schocken. Multilayer feedforward networks with a non-polynomial activation function can approximate any function. *Neural Networks*, 6:861–867, 1993. https://doi.org/10.1016/S0893-6080(05)80131-5.

[29] J.-L. Lions. *Optimal control of systems governed by partial differential equations.* Translated from the French by S. K. Mitter. Die Grundlehren der mathematischen Wissenschaften, Band 170. Springer-Verlag, New York-Berlin, 1971.

[30] Z. Long, Y. Lu, X. Ma, and B. Dong. PDE-Net: Learning PDEs from data. *Proceedings of Machine Learning Research*, 80:3208–3216, 2018. `http://proceedings.mlr.press/v80/long18a.html`.

[31] S. Lu and J. Flemming. Convergence rate analysis of Tikhonov regularization for nonlinear ill-posed problems with noisy operators. *Inverse Problems*, 28:104003, 2012. `https://doi.org/10.1088/0266-5611/28/10/104003`.

[32] D. Ma, V. Gulani, N. Seiberlich, K. Liu, J. Sunshine, J.L. Duerk, and M.A. Griswold. Magnetic resonance fingerprinting. *Nature*, 495(187):187–193, 2013. `https://dx.doi.org/10.1038/nature11971`.

[33] D.J.C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992. `https://doi.org/10.1162/neco.1992.4.3.415`.

[34] G. Mazor, L. Weizman, A. Tal, and Y.C. Eldar. Low-rank magnetic resonance fingerprinting. *Medical Physics*, 45(9):4066–4084, 2018. `https://dx.doi.org/10.1002/mp.13078`.

[35] P. Neittaanmaki, J. Sprekels, and D. Tiba. *Optimization of elliptic systems.* Springer Monographs in Mathematics. Springer, New York, 2006. Theory and applications.

[36] D. Nguyen and B. Widrow. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. *1990 IJCNN International Joint Conference on Neural Networks*, 3:21–26, 1990. `http://dx.doi.org/10.1109/IJCNN.1990.137819`.

[37] A. Pinkus. Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8:143–195, 1999. `https://doi.org/10.1017/S0962492900002919`.

[38] M.J.D. Powell. A view of unconstrained optimization. In L.C.W. Dixon, editor, *Optimization in Action*, pages 117–152. Academic Press, London and New York, 1976.

[39] T. Qin, K. Wu, and D. Xiu. Data driven governing equations approximation using deep neural networks. *Journal of Computational Physics*, 395:620–635, 2019. `https://doi.org/10.1016/j.jcp.2019.06.042`.

[40] D. Ralph. Global convergence of damped Newton's method for nonsmooth equations via the path search. *Mathematics of Operations Research*, 19(2):352–389, 1994. `https://doi.org/10.1287/moor.19.2.352`.

[41] K. Scheffler. A pictorial description of steady-states in rapid magnetic resonance imaging. *Concepts in Magnetic Resonance*, 11(5):187–193, 1999. `https://dx.doi.org/10.1002/(SICI)1099-0534(1999)11:5<291::AID-CMR2>3.0.CO;2-J`.

[42] J. Sirignano and K. Spiliopoulos. DGM: A deep learning algorithm for solving partial differential equations. *Journal of Computational Physics*, 375:1339–1364, 2018. `https://doi.org/10.1016/j.jcp.2018.08.029`.

[43] G. Teschl. *Ordinary Differential Equations and Dynamical Systems*, volume 140 of *Graduate Studies in Mathematics*. American Mathematical Society, first edition, 2012.

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

[45] J. Zowe and S. Kurcyusz. Regularity and stability for the mathematical programming problem in banach spaces. *Applied Mathematics and Optimization*, 5:49–62, 1970. `https://doi.org/10.1007/BF01442543`.