Relaxation approach for learning regularizers by neural networks for a class of optimal control problems

Sébastien Court† Markus Haltmeier‡

September 27, 2021

Abstract

The present paper deals with the data-driven design of regularizers in the form of artificial neural networks, for solving inverse problems formulated as optimal control problems. These regularizers aim at improving accuracy, wellposedness or compensating uncertainties for a class of optimal control problems (inner-problems). Parameterized as neural networks, their weights are chosen in order to reduce a misfit between data and observations of the state solution of the inner-optimal control problems. Learning these weights constitutes the outer-problem. Based on necessary first-order optimality conditions for the inner-problems, a relaxation approach is proposed in order to implement efficient solving of the inner-problems, namely the forward operator of the outer-problem. Optimality conditions are derived for the latter, and numerical illustrations show the feasibility of the relaxation approach, first for rediscovering standard $L^2$-regularizers, and next for designing regularizers that compensate unknown noise on the observed state of the inner-problem.

Keywords: Optimal control theory, Regularization, Neural networks, Identification problem, Data-driven optimization, Learning problems.

AMS subject classifications (2020): 49N45, 49K99, 68T07, 68T05, 68T20, 65J20.

Contents

1 Introduction 2
  1.1 General learning problem ........................................ 2
  1.2 Motivation of data-driven regularization and state-of-the-art .... 3
  1.3 Approach and strategy .......................................... 3
  1.4 Plan ............................................................. 4

2 Notation, assumptions and preliminaries 4
  2.1 Functional setting .............................................. 4
  2.2 The regularizer as realization of a neural network ............ 4
  2.3 General assumptions ........................................... 5
  2.4 On the control-to-state mapping for the inner-problem ....... 5

3 Necessary optimality conditions for the inner-problem 6
  3.1 Differential properties of neural networks ..................... 6
  3.2 Direct approach ............................................... 7
  3.3 Optimal control formalism for the inner-problem (*) ...... 7

*This work is supported by...

†Department of Mathematics & Digital Science Center, University of Innsbruck, Austria, email: sebastien.court@uibk.ac.at.
‡Department of Mathematics, University of Innsbruck, Austria, email: markus.haltmeier@uibk.ac.at.
1 Introduction

Designing regularizers for solving inverse problems is an important data-science problem for which data-driven approaches have been recently developed. In particular, the design of regularizers as artificial neural networks demonstrate interesting capacities for improving characteristics of inverse problems. Among these characteristics, one can aim at recovering uniqueness of solutions for initially ill-posed inverse problems, facilitating their solving, obtaining specific qualitative properties like sparsity, or even compensating unknown real-world noise that leads to uncertainties on the measured data. The design of such regularizers can be achieved towards an optimal control approach, consisting in optimizing parameters of a machine, namely an input-output map, in order to minimize a distance between real-world data and outputs of this machine.

1.1 General learning problem

Consider a series of $K$ problems that belong to a given class of optimal control problems indexed by $1 \leq k \leq K$, and assume that for each of them we know a solution. More precisely, we consider a set of tasks $\hat{z}_k$ that we know to be achieved by the means of a control $\hat{u}_k$. The $K$ pairs $\{(\hat{z}_k, \hat{u}_k), 1 \leq k \leq K\}$ constitutes our data set, and our main goal is, from this data set, to train a regularizer that improves the solving of this class of control problem. Our supervised-learning problem can be formulated as follows:

$$\min_{r \in \mathcal{R}} \frac{1}{2K} \sum_{k=1}^{K} \|u_k - \hat{u}_k\|_2^2,$$

subject to: $u_k$ is solution of Problem (⋆) with $\hat{z}_k, r$, for all $1 \leq k \leq K$,

where Problem (⋆) with $(\hat{z}, r) = (\hat{z}_k, r)$ is given by

$$\min_{(y, u) \in \mathcal{Y} \times \mathcal{U}} (c(y, \hat{z}) + \gamma \circ r(u)), $$

subject to: $\varphi(y, u) = 0$.

The set $\mathcal{R}$ describes the class of regularizers in which we design a – possible – optimal function $r$. The state is denoted by $y$, and the state equation is given by $\varphi(y, u) = 0$. The cost $c(y, \hat{z})$ corresponds to a misfit between
the state $y$ and the task $\hat{z}$ to achieve. The mapping $\gamma$ is given, with values in $\mathbb{R}$, and can correspond for instance to a norm.

1.2 Motivation of data-driven regularization and state-of-the-art

In the formulation of the inner-problem ($\ast$), the additional term $r(u)$ corresponds to a so-called regularizer. Its role consists in forcing the control $u$ to present certain properties, like regularity, boundedness, or sparsity for instance. Besides the qualitative properties aforementioned, designing a regularizer for a class of identification problems can address several issues. For instance, inverse problems are well-known to present non-uniqueness issues. A regularizer designed with a data-driven approach could help to select a relevant solution. An other issue could be due to uncertainty on measurements: Data can be subject to a noise phenomenon whose nature is unknown, and that would lead to inaccuracy on the solution. An appropriate regularizer, implementing implicitly the effects of this noise, could help to compensate this lack of accuracy. We illustrate this last point in our numerical realizations (see section 6).

In order to design such a regularizer, one need to parameterize the set $\mathcal{R}$, that we choose as a set of artificial neural networks (ANNs) with a prescribed number of layers. A regularizer ($u \mapsto r(w, u)) \in \mathcal{R}$ is then a mapping parameterized by a set of weights $w$. We refer to section 3.1 for further details on the definitions and notation related to the ANNs we will consider. Beside the theoretical approximation capacities of ANNs (see [GPEB19] for instance), data-driven methods training ANNs have lead to the design of regularizers that outperform classical solving methods for inverse problems.

For example, the data-driven modeling of regularizers with NETT (Network Tiknonov) was investigated in [LSAH20a], in order to solve inverse problems subject to unknown noise, with applications to Computer Tomography. Let us also cite [OSH19] in the context of trained decoder networks for sparse reconstruction, and refer to [LSAH20b] for a theoretical analysis of such an approach. Further developments were studied in [ONS19, ONSH20], considering aNETT (augmented Network Thikonov), designing a regularizer and also a penalty (misfit) function in the form of a neural network. Still on applications in medical imaging, the use of adversarial neural networks as regularizers was studied in [LOS18]. The use of deep neural networks for solving ill-posed inverse problems related to medical imaging was also studied in [AO17]. The special case of input-convex neural networks (ICNNs) for modeling regularizers with a data-driven approach was considered in [MDS+20]. More generally, we orient the reader to [HN20] for a comprehensive review on techniques related to this topic.

1.3 Approach and strategy

In [HKB18] the authors investigated the problem of learning the Thikhonov regularization parameter for a class of inverse problems, with a bi-level approach. This approach consists in adopting a hierarchy in the formulation of the regularizer learning problem: The identification problems ($\ast$) for which we want to design a regularizer are called inner-problems, and the main problem ($\ast\ast$) consists in optimizing the parameters of the regularizer for a given set of identification problems for which we know the ground-truth (the data set). The bi-level approach was initially developed for image denoising in [KP13].

In the present paper, we adopt this bi-level approach, where the inner-problem ($\ast$) is considered as a constraint of the outer-problem. Compared with [HKB18], in our case the range of possible regularizers is a priori wider than a Thikonov regularizer, as the regularizers are chosen in the form of artificial neural networks parameterized with weights. The price is to pay is a theoretical analysis (existence and uniqueness of minimizers) that becomes out-of-reach in the general case, due in particular to the purely nonlinear aspects of neural networks. Therefore, our main contribution is the development of a relaxation approach, consisting in replacing the optimality constraint of the inner-problems by the corresponding necessary Karush-Kuhn-Tucker conditions. The latter are equality constraints (vanishing gradient equalities) that transform the outer-problem into a standard optimal control problem of state constraint type.

This relaxation approach is inspired by optimal control problems with partial differential equations constraints. Indeed, partial differential equations modeling physical dynamical systems can often be derived from the least action principle. They are merely first-order optimality conditions of a minimization problem. Well-posedness questions for these partial differential equations are then related to existence and uniqueness of minimizers for functionals called actions.

The main theoretical result is given in Theorem 2, where we provide the sensitivity of the objective function of
Problem (**) with respect to the weights of the neural network. This allows us to implement the determination of these weights in Algorithm 1, with gradient rules. Illustrations are proposed by considering as inner-problems the classical identification of coefficients in elliptic partial differential equations. Regularizers are designed, in particular for compensating the effects of a Gaussian noise introduced on the state variable.

1.4 Plan

The paper is organized as follows: Notation and assumptions are given in section 2. In section 3 we study the inner-problems with ANNs as regularizers. The latter are described in section 3.1. The necessary optimality conditions related to the inner-problems are investigated in sections 3.2 and 3.3. The relaxation approach is presented in section 4.1. We prove existence of minimizers for the relaxed outer-problem in section 4.2, and derive optimality conditions in section 4.5 for the latter. In section 5 we present algorithms for implementing the relaxation approach. Section 6 is devoted to numerical illustrations considering the identification of coefficients in an elliptic partial differential equation. Concluding remarks are given in section 7. In Appendix A.1 we present advantages that the optimal control approach can offer when dealing with state-constrained identification problems. Appendix A.2 recalls classical results related to the inverse conductivity problem.

2 Notation, assumptions and preliminaries

2.1 Functional setting

We denote by \( \mathcal{Y}, \mathcal{U}, \mathcal{Q} \) real Banach spaces, and by \( \varphi : \mathcal{Y} \times \mathcal{U} \to \mathcal{Q} \) the state mapping. We assume that \( \mathcal{Q} \) is reflexive. The control space \( \mathcal{U} \) is assumed to be finite-dimensional, of dimension \( n_{in} \), so that \( \mathcal{U} \simeq \mathbb{R}^{n_{in}} \). Therefore in particular \( \mathcal{U} \) is a Hilbert space, and reflexive. The range space \( \mathcal{V} \) of the neural networks is finite-dimensional, of dimension \( n_{out} \), and \( \mathcal{V} \simeq \mathbb{R}^{n_{out}} \). The space of weights is denoted by \( \mathcal{W} \), and described in section 3.1. Given weights \( w \in \mathcal{W} \), we denote by \( r(w, \cdot) : \mathcal{U} \to \mathcal{V} \) the associated neural network of \( \mathcal{R} \), and by \( \gamma : \mathcal{V} \to \mathbb{R} \) a smooth given mapping, possibly chosen as a norm on \( \mathbb{R}^{n_{out}} \).

The data \( \{(\hat{z}_k, \hat{u}_k), 1 \leq k \leq K\} \) are of finite number. The controls \( (\hat{u}_k)_{1 \leq k \leq K} \) are assumed to lie in \( \mathcal{U} \), while the tasks \( (\hat{z}_k)_{1 \leq k \leq K} \) lie in a finite-dimensional subspace of \( \mathcal{Y} \).

2.2 The regularizer as realization of a neural network

Given \( L \geq 1 \), we denote by \( w = (w_1, w_2, \ldots, w_L) \) a vector of affine functions. The set of regularizers, denoted by \( \mathcal{R} \), is made of feedforward neural networks of \( L \) layers, for which the dimensions \( n_\ell \) \((1 \leq \ell \leq L)\) of the hidden layers are prescribed:

\[
\mathcal{R} = \left\{ \mathcal{U} \ni u \mapsto w_L \rho \left( w_{L-1} \rho \left( \ldots w_2 \rho(w_1 u) \right) \right) \in \mathcal{V} \mid (w_1, \ldots, w_L) \in \prod_{\ell=1}^L \text{Aff}(n_\ell, n_{\ell+1}, \mathbb{R}) \right\}
\]

We denoted by \( \text{Aff}(n_\ell, n_{\ell+1}, \mathbb{R}) \) the set of affine functions from \( \mathbb{R}^{n_\ell} \) to \( \mathbb{R}^{n_{\ell+1}} \). For \( 1 \leq \ell \leq L \), its components write

\[
w_\ell : u \mapsto A_\ell u + b_\ell,
\]

where \( A_\ell \in \mathbb{R}^{n_{\ell+1} \times n_\ell} \) and \( b_\ell \in \mathbb{R}^{n_{\ell+1}} \), with \( n_1 = n_{in} = \text{dim}(\mathcal{U}) \) and \( n_{L+1} = n_{out} = \text{dim}(\mathcal{V}) \). For the sake of concision, we will denote throughout the paper

\[
\mathcal{W} := \prod_{\ell=1}^L \text{Aff}(n_\ell, n_{\ell+1}, \mathbb{R}).
\]

We introduce \( \mathcal{L}_W := \sum_{\ell=1}^{L-1} n_\ell \). We endow the space \( \mathcal{W} \) with the Euclidean norm given by

\[
\|w\|^2_{\mathcal{W}} := \sum_{\ell=1}^L \left( \|A_\ell\|^2_{\mathbb{R}^{n_{\ell+1}\times n_\ell}} + |b_\ell|^2_{\mathbb{R}^{n_{\ell+1}}}, \right)
\]

More precisely the sensitivity of the objective function of the relaxed problem (***) introduced in section 4, but the expression of this objective function is the same as the one of Problem (**).
Assume Proposition 1. Stable to define the control-to-state mapping that we denote by $S$ and the neural network $(\text{Assumption (A3))}$ that are solvable. From $(\text{Assumption (A3))}$ we deduce easily existence of minimizers when the regularizer is non-trivial. While addressing the inner-problems only requires first-order differentiability (see section 3), $(\text{Assumption (A3))}$ is formulated as follows:

$$r(w, u) = \begin{cases} w_2(\rho(w_1(u))) & \text{if } L = 2, \\ w_L(\rho(w_{L-1}(\rho(\ldots \rho(w_1(u)))))) & \text{if } L \geq 3. \end{cases}$$

In the expression above, since the functions $w_2$ have values in $\mathbb{R}^{n+1}$, the compositions by $\rho$ have to be understood coordinate-wise as $\rho(w(z))_i = \rho((w_i(z))_i)$, for $z \in \mathbb{R}^{n'}$ and $i \in \{1, \ldots, n_{r+1}\}$. Thus, the regularizer space $\mathcal{R}$ is parameterized by the weights lying in $\mathcal{W}$, and the outer-problem $(\ast \ast)$ is formulated as follows:

$$\begin{align*}
\min_{w \in \mathcal{W}} & \quad \frac{1}{2} \sum_{k=1}^{K} \| \hat{u}_k - u_k \|_{\mathcal{U}}^2, \\
\text{subject to: } & \quad u_k \text{ is solution of Problem } (\ast) \text{ with } (\hat{z}, r) = (\hat{z}_k, r(\cdot, w)), \text{ for all } 1 \leq k \leq K. 
\end{align*}$$

### 2.3 General assumptions

Throughout the paper, we will assume that the following set of assumptions holds:

- **(A1)** The mapping $\varphi$ is of class $C^2$ over $\mathcal{Y} \times \mathcal{U}$. For all $u \in \mathcal{U}$, there exists $y \in \mathcal{Y}$ such that $\varphi(y, u) = 0$.

- **(A2)** For all $(\overline{y}, \overline{u}) \in \mathcal{Y} \times \mathcal{U}$, the linear mapping $\varphi_y'(\overline{y}, \overline{u}) : \mathcal{Y} \to \mathcal{Q}'$ is surjective. Moreover, there exists a constant $C(\overline{y}, \overline{u}) > 0$, depending only on $(\overline{y}, \overline{u})$ such that for all $y \in \mathcal{Y}$ the following estimate holds:

$$\|y\|_{\mathcal{Y}} \leq C \|\varphi_y'(\overline{y}, \overline{u}) y\|_{\mathcal{Q}'}.$$

Instead of (A2), we may consider a weaker assumption, namely (A2)’:

- **(A2)’** For all $(\overline{y}, \overline{u}) \in \mathcal{Y} \times \mathcal{U}$, the linear mapping $\varphi_y'(\overline{y}, \overline{u}) : \mathcal{Y} \to \mathcal{Q}'$ is almost surjective, namely its range is dense in $\mathcal{Q}'$.

- **(A3)** When $r \equiv 0$, Problem $(\ast)$ admits at least one global minimizer.

- **(A4)** The cost function $c : \mathcal{Y} \times \mathcal{U}$ is twice continuously Fréchet-differentiable.

- **(A5)** The activation function $\rho$ is of class $C^2$ on $\mathbb{R}$.

Assumption (A1) combined with (A2) imply the existence of a control-to-state ammping for the inner-problem (see section 2.4). Assumption (A3) is natural, as we aim at designing regularizers for (inner-)problems that are solvable. From (A3) we deduce easily existence of minimizers when the regularizer is non-trivial (Proposition 2). While addressing the inner-problems only requires first-order differentiability (see section 3), the derivation of optimality conditions for the outer-problem (see section 4) necessitates second-order derivatives, in particular for the objective function of the inner-problem, including the cost function $c$ (Assumption (A4)) and the neural network (Assumption (A5)). Throughout the paper these assumptions will be implicitly assumed.

### 2.4 On the control-to-state mapping for the inner-problem

The state constraint is given by the implicit function $\varphi : \mathcal{Y} \times \mathcal{U} \to \mathcal{Q}'$. Under assumptions (A1)-(A2), we are able to define the control-to-state mapping that we denote by $S$:

**Proposition 1.** Assume (A1)-(A2). There exists a mapping $S : \mathcal{U} \to \mathcal{Y}$ of class $C^2$ such that for all $(y, u) \in \mathcal{Y} \times \mathcal{U}$ we have

$$\varphi(y, u) = 0 \iff y = S(u).$$

Furthermore, $S$ is of class $C^2$ over $\mathcal{U}$.

**Proof.** The result is due to the implicit function theorem. \qed

Depending on the context, for the sake of convenience we may consider the original equality constraint $\varphi(y, u) = 0$, or substitute the variable $y$ by $S(u)$. The lack of control-to-state mapping $S$ for the inner-problem would introduce further difficulties that we choose to not address in this paper.
3 Necessary optimality conditions for the inner-problem

Given \( w \in \mathcal{W} \), when the regularizer is chosen in the form of a neural network as described in section 3.1, the inner-problems correspond to the following class of optimal control problems:

\[
\begin{cases}
\min_{(y,u) \in Y \times U} \left( J(y,u,w) := c(y,\hat{z}) + \gamma \circ r(w,u) \right), \\
\text{subject to: } \varphi(y,u) = 0 \text{ in } Q'.
\end{cases}
\]

(\ast)

The aim of this section is to give necessary optimality conditions for the inner-problem (\ast). For that purpose, we first need to investigate the differential properties owned by neural networks of \( \mathcal{W} \).

3.1 Differential properties of neural networks

Consider the partial neural network with \( \ell \) layers (1 \( \leq \ell \leq L \)):

\[
r^{(\ell)}(w^{(\ell)}, u) = \begin{cases} w_1(u) & \text{if } \ell = 1, \\ w_2(\rho(w_1(u))) & \text{if } \ell = 2, \\ w_2(\rho(\rho(w_1(u)))) & \text{if } \ell \geq 3. \end{cases}
\]

Recall that the weights \( (w_{\ell}) \) are affine functions of \( \mathbb{R}^{n_{\ell}} \), of the form \( w_{\ell}(z) = A_{\ell}z + b_{\ell} \) with \( A_{\ell} \in \mathbb{R}^{n_{\ell+1} \times n_{\ell}} \) and \( b_{\ell} \in \mathbb{R}^{n_{\ell+1}} \). We have denoted

\[
w^{(\ell)} = (w_1, \ldots, w_\ell) \in \prod_{k=1}^{\ell} \text{Aff}(\mathbb{R}^{n_k}, \mathbb{R}^{n_{k+1}}).
\]

Observe that for 1 \( \leq \ell \leq L - 1 \) the following formula holds:

\[
r^{(\ell+1)}(w^{(\ell+1)}, u) = w_{\ell+1}\rho\left(r^{(\ell)}(w^{(\ell)}, u)\right).
\]

(1)

From (1), by induction we can verify that the sensitivity of \( r^{(L)} \) with respect to \( u \) satisfies the identity

\[
\frac{\partial r^{(L)}}{\partial u}(w^{(L)}, u) = A_L \prod_{\ell=1}^{L-1} \rho'(r^{(\ell)}(w^{(\ell)}, u)) A_{\ell}.
\]

(2)

The vector/matrix product \( \rho'(r^{(\ell)}(w^{(\ell)}, u)) A_{\ell} \) is a matrix, and has to be understood as follows:

\[
(\rho' A)_{ij} = \rho'_{,ij} A_{ij}.
\]

(3)

The symbol \( \prod_{\ell=1}^{L-1} \) is used for calculating the non-commutative product of matrices, by multiplying them iteratively to the left when the index \( \ell \) increases: \( \prod_{\ell=1}^{L} B_{\ell} = B_{L} B_{L-1} \ldots B_2 B_1 \). Without ambiguity, we will simply denote \( r = r^{(L)} \) and \( w = w^{(L)} \). Using the mean-value theorem, by induction we can estimate

\[
\|r(w,u) - r(w,0)\|_{\mathcal{U}} \leq \left( \prod_{\ell=1}^{L} \|A_{\ell}\|_{\mathbb{R}^{n_{\ell+1} \times n_{\ell}}} \right) \|\rho'\|_{\infty} \|u\|_{\mathcal{U}},
\]

\[
\|r(w,u_1) - r(w,u_2)\|_{\mathcal{U}} \leq \left( \prod_{\ell=1}^{L} \|A_{\ell}\|_{\mathbb{R}^{n_{\ell+1} \times n_{\ell}}} \right) \|\rho'\|_{\infty} \|u_1 - u_2\|_{\mathcal{U}},
\]

where we recall \( \mathcal{L}_{\mathcal{W}} = \sum_{\ell=1}^{L-1} n_{\ell} \). The differentiation of \( r \) with respect to the weights \( w_{\ell} \) is given by the following formula, for all \( \tilde{w} \in \text{Aff}(\mathbb{R}^{n_{\ell}}, \mathbb{R}^{n_{\ell+1}}) \):

\[
\frac{\partial r}{\partial w_{\ell}}(w,u)\tilde{w} = A_L \left( \prod_{k=\ell}^{L-1} \rho'(r^{(k)}(w^{(k)}, u)) A_{k} \right) \left( \tilde{w}(\rho(r^{(\ell-1)}(w^{(\ell-1)}, u))) \right).
\]
We can obtain this formula also by induction, using (1). Here the product has to be understood as follows for
1 ≤ i ≤ nout:
\[
\left( \frac{\partial r}{\partial w_i}(w,u) \right) = (A_L)_{ij} \left( \prod_{k=l}^{L-1} \left( \rho^j \left( r^{(k)}(w^{(k)}), u \right) A_k \right) \right)_{j,i}.
\]

These expressions for the derivatives of the neural network will be used in practice for the implementation of
the gradient associated with the inner-problem. Let us now derive expressions for the latter.

### 3.2 Direct approach

The control-to-state mapping \( S \) obtained in Proposition 1 enables us to derive classically necessary optimality
conditions for the inner-problems directly by omitting the state constraint. Indeed, using \( y = S(u) \), and given
\( w \in W \), we can rewrite Problem (*) as
\[
\min_{u \in U} \left( J(u, w) := J(S(u), u, w) = c(S(u), z) + \gamma(r(w, u)) \right).
\]

Using Assumption (A3) and the formulation above, we obtain existence of minimizers for the inner-problem (*):

**Proposition 2.** Under Assumption (A3), for all \( w \in W \) Problem (*) admits a global minimizer.

**Proof.** From Assumption (A3) and Proposition 1, the mapping \( u \mapsto c(S(u), z) \) admits global minimizers, and
due to the regularity of \( u \mapsto \gamma(r(w, u)) \), the result follows straightforwardly.

We next obtain necessary optimality conditions involving the derivative of \( S \):

**Proposition 3.** Let be \( w \in W \). If \((y = S(u), u) \in Y \times U\) is solution of Problem (*), then the gradient
\[
G(u, w, z) := \frac{\partial J}{\partial u}(u, w) = S' (u)^* c_y(S(u), z) + r'_u(w, u)^* \gamma'(r(w, u))
\]
vanishes.

**Proof.** Following the previous comments and Proposition 1, the result follows from the Karush-Kuhn-Tucker
conditions.

In certain problems, like shape optimization for example, evaluating efficiently values of the control-to-state
mapping \( S(u) \), and in particular \( S'(u)^* \), can be challenging numerically, or expensive computationally. Therefore
we prefer sometimes to adopt an optimal control approach, also called the adjoint approach, that we develop
in our context in section 3.3 below. See remarks in [HPUU09, section 1.6.1, page 59] for further explanations.
Further comments on the possible advantages of the optimal control approach are given in Appendix A.1.

### 3.3 Optimal control formalism for the inner-problem (*)

Let us recall the initial description of the inner-problem, when we keep the original state constraint:

\[
\left\{ \begin{array}{l}
\min_{(y, u) \in Y \times U} \left( J(y, u, w) := c(y, z) + \gamma \circ r(w, u) \right), \\
\text{subject to: } \varphi(y, u) = 0 \text{ in } Q'.
\end{array} \right.
\]

Unlike the direct approach which utilizes \( \varphi(y, u) = 0 \Leftrightarrow y = S(u) \), the optimal control approach – also called
the adjoint approach – consists in taking into account the state constraint by introducing a Lagrange multiplier.
We define the Lagrangian of problem (*) as follows
\[
\mathcal{L} : (Y \times U) \times Q \rightarrow \mathbb{R} \\
((y, u), p) \mapsto c(y, z) + \gamma \circ r(w, u) + \langle p, \varphi(y, u) \rangle_{Q', Q'}.
\]

Problem (*) can be solved by finding a saddle-point to the Lagrangian \( \mathcal{L} \), minimized with respect to variables
\( (y, u) \) and maximizing with respect to \( p \). The sensitivity of \( \mathcal{L} \) with respect to variable \( y \) yields the adjoint
equation, whose unknown is the adjoint state \( p \in Q \). It is given in \( Y' \) by
\[
\varphi'_{y}(y, u)^* p + c'_{y}(y, z) = 0.
\]

We define a solution of system (5) by transposition.
**Proposition 5.** Let be problem, namely the quantity
\[ \varphi'(y, u) \hat{y} = f, \]  
where \( \hat{y} \in \mathcal{Y} \) denotes the solution of the following linear system in \( \mathcal{Q}' \):

\[ \varphi'(y, u) \hat{y} = f. \]  

From assumption \((\mathbf{A2})\), system \((7)\) is well-posed. The existence of solutions for adjoint system \((5)\), in the sense of Definition 1, is stated as follows:

**Proposition 4.** Let be \((y, u) \in \mathcal{Y} \times \mathcal{U}\). There exists a unique solution \( p \in \mathcal{Q} \) to system \((5)\), in the sense of Definition 1, and there exists a constant \( C(y, u) > 0 \) (depending only on \((y, u)\)) such that
\[ \|p\|_{\mathcal{Q}} \leq C(y, u)\|c'(y, \hat{z})\|_{\mathcal{Y}'}. \]

**Proof.** From assumption \((\mathbf{A2}')\), it is easy to verify that the mapping \( \varphi(y, u)^* \) is injective, and so uniqueness holds. Let us prove existence. Denote by \( \Lambda(y, u) : \mathcal{Q}' \to \mathcal{Y} \) the linear operator which maps \( f \) to \( \hat{y} \), where \( \hat{y} \) is the solution of \((7)\). From Assumption \((\mathbf{A2})\), the operator \( \Lambda(y, u) \) is well-defined and bounded, and consequently \( \Lambda(y, u)^* : \mathcal{Y}' \to \mathcal{Q}' \simeq \mathcal{Q} \) too. Now define \( p = -\Lambda(y, u)^*c'(y, \hat{z}) \). Then, for all \( f \in \mathcal{Q}' \), we verify that
\[ \langle p, f \rangle_{\mathcal{Q}, \mathcal{Q}'} = -\langle c'(y, \hat{z}), \Lambda(y, u)f \rangle_{\mathcal{Y}', \mathcal{Y}} = -\langle c'(y, \hat{z}), z \rangle_{\mathcal{Y}', \mathcal{Y}} = -c'(y, \hat{z}), z, \]
and the announced estimate follows from boundedness of \( \Lambda(y, u)^* \), which completes the proof, as uniqueness is straightforward.

Consequently, the adjoint approach provides another expression for the gradient associated with the inner-problem, namely the quantity \( G(u, \hat{z}) \) introduced in \((4)\).

**Proposition 5.** Let be \( w \in \mathcal{W} \). If \( u \) is solution to Problem \((*)\), then the gradient \( G(u, w, \hat{z}) \) introduced in Proposition 3 writes
\[ G(u, w, \hat{z}) = r'_u(w, u)^*\gamma'(r(w, u)) + \varphi'(y, u)^*p, \]
where \( y = \mathcal{S}(u) \), \( p \) is the solution of \((5)\) corresponding to \((y, u)\), and necessarily \( G(u, w) = 0 \).

**Proof.** From Proposition 1, Problem \((*)\) reduces to \( \min_{u \in \mathcal{U}} J(\mathcal{S}(u), u, w) \). Following the Karush-Kuhn-Tucker conditions, if \( u \) is optimal to Problem \((*)\), then \( \frac{\partial}{\partial u} (J(\mathcal{S}(u), u, w)) = 0 \), with for all \( \tilde{u} \in \mathcal{U} \)
\[ \frac{\partial}{\partial u} (J(\mathcal{S}(u), u, w)) \cdot \tilde{u} = J'_u(\mathcal{S}(u), u, w)(\mathcal{S}'(u) \tilde{u}) + J'_u(\mathcal{S}(u), u, w) \tilde{u} \]
\[ = \langle c'(y, \hat{z}), \mathcal{S}'(u) \tilde{u} \rangle_{\mathcal{U}} + \langle \gamma'(r(w, u)), r'_u(w, u) \rangle_{\mathcal{Y}} \]
\[ = \langle \mathcal{S}'(u)^*c'(y, \hat{z}) + r'_u(w, u)^*\gamma'(r(w, u)), \tilde{u} \rangle_{\mathcal{U}} \]
\[ = -\mathcal{S}'(u)^*\varphi'(y, u)^*p + r'_u(w, u)^*\gamma'(r(w, u)), \tilde{u} \rangle_{\mathcal{U}}, \]
where \( y = \mathcal{S}(u) \) and \( p \) satisfies \((5)\). By differentiating the identity \( \varphi(\mathcal{S}(u), u) = 0 \), for all \( \tilde{u} \in \mathcal{U} \) we obtain
\[ \varphi'(y, u) \tilde{u} + \varphi'(y, u) = 0, \]
which implies that
\[ -\mathcal{S}'(u)^*\varphi'(y, u)^* = \varphi'(y, u)^*, \]
and thus \((9)\) reduces to
\[ \frac{\partial}{\partial u} (J(\mathcal{S}(u), u, w)) \cdot \tilde{u} = \langle \varphi'(y, u)^*p + r'_u(w, u)^*\gamma'(r(w, u)), \tilde{u} \rangle_{\mathcal{U}}, \tilde{u} = 0 \]
for all \( \tilde{u} \in \mathcal{U} \), which completes the proof. \( \square \)
Remark 1. One could also have obtained the necessary conditions of Proposition 5 by considering a stationary point of the Lagrangian functional $\mathcal{L}$. Indeed, vanishing the derivative of $\mathcal{L}$ with respect to $p$ yields $\varphi(y, u) = 0 \iff y = \mathbb{S}(u)$, in virtue of Proposition 1. Vanishing the derivative of $\mathcal{L}$ with respect to $y$ leads to the adjoint system (5). With the chain rule, we calculate for all $\tilde{u} \in \mathcal{U}$:

$$
\frac{\partial \mathcal{L}}{\partial u}(w, u, \tilde{u}) = \langle \gamma'(r(w, u)), r'_u(w, u) \tilde{u} \rangle + c'_y(S(u), \tilde{u}),
$$

$$
= \langle r'_u(w, u) \gamma'(r(w, u)), \tilde{u} \rangle_{\mathcal{U}} - \langle \varphi'_y(S(u), \tilde{u}) \gamma'(r(w, u)), \mathbb{S}(\tilde{u}) \rangle_{\mathcal{V}} = 0.
$$

(11)

where $p$ is solution of (5) with $y = \mathbb{S}(u)$. Using (10), we see that (11) reduces to

$$
\frac{\partial \mathcal{L}}{\partial u}(w, u) = (r'_u(w, u) \gamma'(r(w, u)), \tilde{u})_{\mathcal{U}} + \langle p, \varphi'_u(y, u, \tilde{u}) \rangle_{\mathcal{U}'} = \langle G(u, w, \tilde{u}), \tilde{u} \rangle_{\mathcal{U}'}.
$$

which leads to the same result as Proposition 5. Therefore the identities given in Proposition 5 can be summarized as

$$
\left( \frac{\partial \mathcal{L}}{\partial y}(y, u, p), \frac{\partial \mathcal{L}}{\partial u}(y, u, p), \frac{\partial \mathcal{L}}{\partial p}(y, u, p) \right) = 0.
$$

(12)

The necessary optimality conditions given in Proposition 3 are self-consistent, and thus more appropriate for theoretical analysis, while those provided by Proposition 5 are more convenient for numerical realization. Following these two results, the constraint of Problem (**) is then relaxed, by being simply replaced by the identity $G(u, w, \tilde{z}) = 0$.

4 Relaxation approach for the outer-problem using necessary optimality conditions of the inner-problems

4.1 Relaxed formulation of the main problem

The aim of the present section is to translate the abstract constraint of Problem (***), namely "$u_k$ is solution of Problem (**)", into an equality constraint. For that purpose, we use the necessary vanishing gradient condition given in Propositions 3 and 5. Note that a solution to Problem (*) with $(\tilde{z}, r) = (z_k, r(w, \cdot))$ satisfies necessarily $G(u_k, w, \tilde{z}_k) = 0$, but if it satisfies this identity only, then in general this is not necessarily a solution of Problem (*). Therefore we consider the relaxed version of (**) below:

$$
\begin{align*}
\min_{w \in \mathcal{W}} & \quad \frac{1}{2K} \sum_{k=1}^{K} \| u_k - \hat{u}_k \|_{\mathcal{U}}^2 + \frac{\nu}{2} \| w \|_{\mathcal{V}}^2, \\
\text{subject to: } & \quad G(u_k, w, \hat{z}_k) = 0 \quad \text{for all } 1 \leq k \leq K.
\end{align*}
$$

(***)

The parameter $\nu \geq 0$ is given, and introduced for theoretical purpose mainly (see Theorem 1). This regularization term for the set of weights $w$ could also facilitate the numerical solving of Problem (**), by forcing the weights to be bounded. Note that a parameter-to-solution mapping $w \mapsto u$ characterizing $G(u, w, \tilde{z}) = 0$ is not necessarily available. Even if such a mapping exists, its derivative is difficult to describe explicitly. See section 4.4 for more details. For taking into account the equality constraints of problem (**), we adopt an adjoint approach and introduce the multipliers $\mu \in \mathcal{U}$, with the following Lagrangian mapping:

$$
\mathbf{L} : \quad \mathcal{U} \times \mathcal{W} \times \mathcal{U} \to \mathbb{R}
$$

$$
(u, w, \mu) \mapsto \frac{1}{2} \| u - \tilde{u} \|_{\mathcal{U}}^2 + \frac{\nu}{2} \| w \|_{\mathcal{V}}^2 + \langle \mu, G(u, w, \tilde{z}) \rangle_{\mathcal{U} \times \mathcal{U}'}.
$$

(13)

Like in section 3.3 for the inner-problems (see Remark 1), a solution of Problem (***) can be sought – when $K = 1$ – as a saddle-point of $\mathbf{L}$ with respect to the variables $(u, w)$ and $\mu$. In practice, we shall consider $(u_k)_{1 \leq k \leq K}$, $\mu = (\mu_k)_{1 \leq k \leq K}$ and

$$
\mathbf{L}^{(K)} : \quad \mathcal{U}^K \times \mathcal{W} \times \mathcal{U}^K \to \mathbb{R}
$$

$$
(u, w, \mu) \mapsto \frac{1}{2K} \sum_{k=1}^{K} \| u_k - \hat{u}_k \|_{\mathcal{U}}^2 + \frac{\nu}{2} \| w \|_{\mathcal{V}}^2 + \frac{1}{K} \sum_{k=1}^{K} \langle \mu_k, G(u_k, w, \hat{z}_k) \rangle_{\mathcal{U} \times \mathcal{U}'}.
$$

(14)
Up to a change of the functional framework, for the theoretical questions we can simply consider the functional introduced in (13), by considering the vector version of the different variables, namely \( u = (u_k)_{1 \leq k \leq K}, \tilde{u} = (\tilde{u}_k)_{1 \leq k \leq K} \) and \( \mu = (\mu_k)_{1 \leq k \leq K} \). Before deriving conditions of stationarity for L, let us discuss about existence of solutions for Problem (\( \ast \ast \)).

### 4.2 Existence of minimizers for the relaxed problem

Existence of solutions for Problem (\( \ast \ast \)) follows from standard arguments:

**Theorem 1.** Assume that \( \nu > 0 \). Then Problem (\( \ast \ast \)) admits a global minimizer.

**Proof.** The existence of feasible solutions for Problem (\( \ast \ast \)) is due to Proposition 2, combined with Proposition 3 (or also Proposition 5). Since the functional

\[
\mathcal{U} \times \mathcal{W} \ni (u, w) \mapsto \frac{1}{2} \| u - \tilde{u} \|^2_u + \frac{\nu}{2} \| w \|^2_w
\]

is bounded from below, Problem (\( \ast \ast \)) admits a minimizing sequence \((u^{(n)}, w^{(n)})_n\) of feasible solutions. Further, from (15), the sequences \((u^{(n)})_n\) and \((w^{(n)})_n\) are bounded in \( \mathcal{U} \) and \( \mathcal{W} \), respectively. Remind that the spaces \( \mathcal{U} \) and \( \mathcal{W} \) are finite-dimensional. Consequently, up to extraction, there exists \((u, w) \in \mathcal{U} \times \mathcal{W}\) such that \((u^{(n)}, w^{(n)})_n\) converges strongly towards \((u, w)\). It is clear that \( \mathcal{U} \times \mathcal{W} \ni (u, w) \mapsto G(u, w, \tilde{z}) \in \mathcal{U} \) is continuous, so that we can pass to the limit in \( G(u^{(n)}, w^{(n)}, \tilde{z}) = 0 \) for obtaining \( G(u, w, \tilde{z}) = 0 \), which concludes the proof, as \((u, w)\) is then solution of Problem (\( \ast \ast \)). \( \square \)

**Remark 2.** If we consider the case \( \nu = 0 \), the boundedness of the sequence \((w_n)_n\) in the proof above is a priori not guaranteed. For proceeding like we did, in such a case we may need to assume further hypotheses on the neural network. For example, we could exploit the equality

\[
S'(u^{(n)}) \gamma_p(S(u^{(n)}), \tilde{z}) + r'_{u'}(w^{(n)}, u^{(n)}) \gamma'(r(w^{(n)}, u^{(n)})) = 0,
\]

due to the identity (4). The sequence \((u^{(n)})_n\) still converges, which implies that the quantity

\[
r_{u'}(w^{(n)}, u^{(n)}) \gamma'(r(w^{(n)}, u^{(n)}))
\]

is uniformly bounded with respect to \( n \). In the simple one-dimensional case where \( n_{in} = n_{out} = 1 \), and \( \gamma \equiv \text{Id} \), this would lead us to consider that the boundedness of the gradient of the neural network implies the boundedness of its weights, which is not true in general. Therefore considering \( \nu > 0 \) seems to be reasonable, as this provides a simple proof to the existence of minimizers.

### 4.3 Derivatives of second-order for the neural network

In section 4.5, we will need expressions of the second-order derivatives of the neural network. From the formula (2) obtained in section 3.1, we derive

\[
r''_{uu}(w, u) = A_L \sum_{\ell = 1}^{L - 1} \left( \prod_{k = 1}^{\ell - 1} \rho'(r^{(k)}) A_k \right) \rho''(r^{(\ell)}) \frac{\partial r^{(\ell)}}{\partial u} A_L \left( \prod_{k = \ell + 1}^{L - 1} \rho'(r^{(k)}) A_k \right),
\]

and

\[
r''_{uws}(w, u, \tilde{w}) = A_L \sum_{\ell = 1}^{L - 1} \left( \prod_{k = 1}^{\ell - 1} \rho'(r^{(k)}) A_k \right) \frac{\partial}{\partial w_s} \left( \rho(r^{(\ell)}(w^{(\ell)}), u) A_{\ell + 1} \right) \tilde{w} \left( \prod_{k = \ell + 1}^{L - 1} \rho'(r^{(k)}) A_k \right)
\]

for all \( 1 \leq s \leq L - 1 \), where we have simply denoted \( r^{(k)} = r^{(k)}(w^{(k)}, u) \). We recall

\[
\frac{\partial r^{(\ell)}}{\partial u}(w^{(\ell)}, u) = A_L \prod_{k = 1}^{\ell - 1} \rho'(r^{(k)}) A_k, \quad \frac{\partial r^{(\ell)}}{\partial w_s}(w^{(\ell)}, u, \tilde{w}) = A_L \left( \prod_{k = s}^{\ell - 1} \rho'(r^{(k)}) A_k \right) \tilde{w}(\rho(r^{(s - 1)}))
\]
and for all \( \tilde{w} : z \mapsto \tilde{A}z + \tilde{b} \)

\[
\frac{\partial}{\partial w_s} \left( \rho'(r^{(\ell)}(w^{(\ell)}, u)) \tilde{A} \right) \cdot \tilde{w} = \begin{cases} 
0 & \text{if } s > \ell, \\
\rho'(r^{(\ell)}(w^{(\ell)}, u)) \tilde{A} + \rho''(r^{(\ell)}) \left( \tilde{w} \rho(r^{(\ell-1)}) \right) \tilde{A} \ell & \text{if } s = \ell, \\
\rho''(r^{(\ell)}) \left( \frac{\partial r^{(\ell)}}{\partial w_s} \right) \tilde{A} \ell & \text{if } s < \ell.
\end{cases}
\]

Like for (3), the products of type \( (\rho''\tilde{r})_i = \rho'' \tilde{r}_i \) are defined componentwise. For \( s = L \), we have for all \( \tilde{w} : z \mapsto \tilde{A}z + \tilde{b} \)

\[
r''_{uwL}(w, u) \cdot \tilde{w} = \prod_{k=1}^{\ell-1} \rho'(r^{(k)}) A_k.
\]

These expressions in themselves are not essential for what follows in the present section, but they can help the reader to reproduce the numerical results we obtained in section 6.

### 4.4 Properties of the adjoint operator

The adjoint equation for Problem \((\hat{\star})\) is obtained by differentiating the Lagrangian \( L \) with respect to the variable \( u \), which gives

\[
G'_u(u, w, \hat{z})^* \cdot \mu = -(u - \hat{u}). \tag{18}
\]

Let us take a look at the operator \( G'_u(u, w, \hat{z})^* \in \mathcal{L}(U, \mathcal{U}') \) which arises in this equation. Since \( \mathcal{U} \simeq \mathbb{R}^{n_m} \) is assumed to be of finite dimension, for each \( (u, w) \in \mathcal{U} \times \mathcal{W} \) the operator \( G'_u(u, w, \hat{z}) \) can be represented by a matrix of \( \mathbb{R}^{n_m \times n_m} \). Further, reminding that \( G(u, w, \hat{z}) = \frac{\partial}{\partial u} J(S(u), u, \hat{w}) \), it yields that \( G'_u(u, w, \hat{z}) \) is an Hessian matrix, and so it is represented by a symmetric matrix. Let us give the expression of \( G'_u(u, w, \hat{z}) \), by denoting \( y = S(u) : \)

\[
G'_u(u, w, \hat{z}) = S'(u) c''_{wp}(y, \hat{z}) S'(u) + c'_p(y, \hat{z}) S''(u) + \gamma'(r(w, u)) r''_{uw}(w, u) + r'(w, u)^* \gamma''(r(w, u)) r'(w, u).
\]

More specifically, for all \( v_1, v_2 \in \mathcal{U} \) we have:

\[
\langle G'_u(u, w, \hat{z}).v_1; v_2 \rangle_{\mathcal{U}', \mathcal{U}} = c''_{wp}(y, \hat{z}).(S'(u).v_1, S'(u).v_2) + c'_p(y, \hat{z}).(S''(u).(v_1, v_2)) + \gamma'(r(w, u))(r''_{uw}(w, u).v_1, r'(w, u).v_2) + \gamma'(r(w, u))(r''_{uw}(w, u).(v_1, v_2)).
\]

Solving equation (18) determines the value of the variable \( \mu \), and thus the question arises whether the operator \( G'_u(u, w, \hat{z})^* = G'_u(u, w, \hat{z}) \) is invertible. Let us comment on the general case where \( G'_u(u, w, \hat{z}) \) is not necessarily invertible.

#### 4.4.1 When the adjoint operator is invertible

Consider any \( (u, w) \in \mathcal{U} \times \mathcal{W} \) such that \( G'_u(u, w, \hat{z}) \) is an invertible matrix. Analogously to the inner-problems, such a case corresponds to the assumption \((A2)\), and following section 2.4, we can prove via the implicit function theorem the existence of a mapping \( w \mapsto u \) characterizing the equality \( G(u, w, \hat{z}) = 0 \). This is a mapping of type control-to-state, since for the outer-problem the variable \( u \) plays the role of a state and \( w \) is the command. This case corresponds to variables \( w \in \mathcal{W} \) such that the associated inner-problem admits a unique solution \( u \in \mathcal{W} \).

Since \( G'_u(u, w, \hat{z}) \) is a square matrix, the matrix \( G'_u(u, w, \hat{z})^* \) is also invertible, and so the adjoint equation (18), namely \( G'_u(u, w, \hat{z})^* \mu + (u - \hat{u}) = 0 \) in the generic case, admits the unique solution

\[
\mu = -G'_u(u, w, \hat{z})^{-T}(u - \hat{u}).
\]

Therefore in this case we are able to derive the necessary Karush-Kuhn-Tucker conditions for Problem \((\hat{\star})\), as the associated qualification constraints are satisfied without ambiguity (see Theorem 2).
4.4.2 When the adjoint operator is not invertible

In the case where \( G_u'(u, w, \hat{z}) \) is not invertible, we introduce a perturbation of the latter, based on the following basic result:

**Lemma 1.** Let \( A \) be a matrix. There exists \( \varepsilon > 0 \) such that \( A - \varepsilon I \) is invertible.

**Proof.** Denote by \( \chi(\lambda) := \det(A - \lambda I) \) the characteristic polynomial of \( A \). If \( \text{Sp}(A) = \{0\} \), any \( \varepsilon \neq 0 \) is suitable. Otherwise, choose any \( \varepsilon \) such that \( 0 < \varepsilon < \inf \{ |\lambda|, \lambda \in \text{Sp}(A) \setminus \{0\} \} \). Then we have \( \det(A - \varepsilon I) = \chi(\varepsilon) \neq 0 \), which concludes the proof.

Therefore, when \( G_u'(u, w, \hat{z}) \) is singular, we can find \( \varepsilon(u, w, \hat{z}) > 0 \) small enough such that \( G_u'(u, w, \hat{z}) - \varepsilon(u, w, \hat{z})I \) is invertible. Note that, up to adapting the proof above, we can also choose \( \varepsilon(u, w, \hat{z}) > 0 \) small enough such that \( G_u'(u, w, \hat{z}) + \varepsilon(u, w, \hat{z})I \) is invertible. Thus we replace \( G_u'(u, w, \hat{z}) \) by \( G_u'(u, w, \hat{z}) \pm \varepsilon(u, w, \hat{z})I \). This is equivalent to add the small \( L^2 \)-type regularizer \( u \mapsto \frac{\varepsilon}{2} \| u \|_U^2 \) to the functional of the inner-problem \((*)\).

Note that the data \( \hat{z} \) are of finite number, and thus the corresponding solutions \( u_k \) of the inner-problems are of finite number too. Therefore \( \varepsilon > 0 \) can be chosen independent of \( u \) and \( \hat{z} \). In practice, we can choose \( \varepsilon = \varepsilon(w) > 0 \) as small as desired, as long as \( G_u'(u, w, \hat{z}) \pm \varepsilon(w)I \) is invertible.

4.5 Necessary optimality conditions for the relaxed problem

The following result gives necessary conditions that an optimal solution of Problem \((\ast \ast)\) has to fulfill.

**Theorem 2.** Assume that \( w \in W \) is a solution of Problem \((\ast \ast)\). Then for all \( 1 \leq k \leq K \) we have \( G(u_k, w, \hat{z}_k) = 0 \), and necessarily

\[
G_\ell(w) := \nu w_\ell + \frac{1}{K} \sum_{k=1}^K G'_{w_\ell}(u_k, w, \hat{z}_k)^* \mu_k = 0
\]  

(19)

for all \( 1 \leq \ell \leq L \), where \( \mu_k \) is the solution of

\[
G'_u(u_k, w, \hat{z}_k)^* \mu_k + (u_k - \hat{u}_k) = 0
\]  

(20)

for each \( 1 \leq k \leq K \).

**Proof.** Since the linear mappings \( G'_u(u_k, w, \hat{z}_k) \in \mathcal{L}(U, U') \) are invertible for each \( 1 \leq k \leq K \), or can be perturbed such that they become invertible (see section 4.4.2), the qualification constraints of the Karush-Kuhn-Tucker conditions are satisfied for the Lagrangian (14). Therefore a solution of Problem \((\ast \ast)\) is necessary a critical point of \( \mathsf{L}^{(K)} \), which leads to the announced identities.

In practice, we solve Problem \((\ast \ast)\) iteratively, by updating the weights \( w \) such that the norm of (19) converges to zero. Given a set of weights \( w \), we first compute each \( u_k \) solution of \((*)\) corresponding to \((\hat{z}, r) = (\hat{z}_k, r(w, \cdot))\). Note that the variable \( u = (u_k)_{1 \leq k \leq K} \) is a state variable in \((\ast \ast)\). Next we compute \( \mu = (\mu_k)_{1 \leq k \leq K} \), where each \( \mu_k \) is solution of (20). Finally we are able to determine the values of the left-hand-sides \( G_\ell(w) \) in (19), that we use in a gradient rule.
5 A gradient-based algorithm

This section is devoted to algorithmic implementation of the relaxed approach for Problem (**). Let us first explain how the inner-problems (*) can be solved in practice.

5.1 Nesterov gradient descent for solving the inner-problems

We solve the inner-problems with Nesterov algorithm [Nes83], also called the accelerated gradient method. It offers a nice compromise between robustness and rapidity of convergence, as inner-problems need to be solved many times, when solving the outer-problem. The corresponding method is given in Algorithm 0 below.

```
Algorithm 0 Solving the inner-problem for a given regularizer \( u \mapsto r(u, w) \) via first-order necessary optimality conditions.

Initialization:
- Initialize \( u^{(0)} \) as \( \frac{1}{K} \sum_{k=1}^{K} \hat{u}_k \).

Initial gradient: From \( u^{(0)} \), compute the (initial) gradient as follows:
  - Compute the state \( y \) corresponding to \( u^{(0)} \), by solving \( \varphi(y, u^{(0)}) = 0 \).
  - Compute the costate \( p \) as solution of \( \varphi'(y, u)^*p = -c'(y, \hat{z}) \).
  - Compute the gradient \( G^{(0)} = G(u, w, \hat{z}) \).

Armijo rule: Choose \( \alpha = 0.5 \).
  - Find the smallest \( n \in \mathbb{N} \) such that \( J(u^{(0)} - \alpha^n G^{(0)}) < J(u^{(0)}) \).
  - Define \( u^{(1)} = u^{(0)} - \alpha^n G^{(0)} \).
  - Compute the gradient \( G^{(1)} \) as above, corresponding to \( u^{(1)} \).
  - Store \( u^{(1)} \) and \( G^{(1)} \).

Nesterov gradient steps: Initialization with \( (u^{(0)}, G^{(0)}) \) and \( (w^{(1)}, G^{(1)}) \).
  Compute iteratively \( u^{(n)} (n \geq 2) \) with the Nesterov steps.
  While \( ||G(u^{(n)}, w, \hat{z})||_\ell > 1.e^{-10} \), do gradient steps.

End: Obtain \( u \), approximated solution of (*) with \( (\hat{z}, r) = (\hat{z}, r(w, \cdot)) \).
```

Note that for evaluating (19), Algorithm 0 needs to be performed \( K \) times, since we need to determine each \( u_k \) corresponding to \( \hat{z}_k \).
5.2 Barzilai-Borwein gradient rule for the relaxed outer-problem

We solve the outer-problem with the Barzilai-Borwein algorithm [Ray97]. The corresponding method is explained in Algorithm 1, choosing $\nu = 0$.

Algorithm 1 Solving the first-order optimality conditions for the outer-problem.

Initialization:
- Load/create the data set.
- Choose random weights as initial weights $w^{(0)}$ for the neural network.

Initial gradient: From $w^{(0)}$, compute the (initial) gradient. For each $1 \leq k \leq K$:
- With Algorithm 0, find the approximated solution $u_k$ of the inner-problem ($\ast$) corresponding to $(\hat{z}, r) = (\hat{z}_k, r(w^{(0)}))$.
- Compute the costate $\mu_k$ as solution of $G^*_w(u_k, w^{(0)}, \hat{z}_k)^* \mu_k = -(u_k - \hat{u}_k)$.
- Compute the gradient $G^{(0)}_\ell = G^*_w(u_k, w^{(0)}, \hat{z}_k)^* \mu_k$.

Compute the gradient in 3 steps

Store $w^{(0)}$ and $G^{(0)} = (G^{(0)}_1, \ldots, G^{(0)}_L)$.

Armijo rule: Choose $\beta = 0.5$.
- Find the smallest $n \in \mathbb{N}$ such that $J(w^{(0)} - \beta^n G^{(0)}) < J(w^{(0)})$.
- Define $w^{(1)} = w^{(0)} - \beta^n G^{(0)}$.
- Compute the gradient $G^{(1)}$ as above, corresponding to $w^{(1)}$.
- Store $w^{(1)}$ and $G^{(1)}$.

Barzilai-Borwein steps: Initialization with $(w^{(0)}, G^{(0)})$ and $(w^{(1)}, G^{(1)})$.

Compute iteratively $w^{(n)}$ ($n \geq 2$) with the Barzilai-Borwein steps.
While $\|G(w^{(n)})\|_W > 1.e^{-8}$, do gradient steps.

End: Obtain $w \in \mathcal{W}$, approximated solution of ($\ast \ast$).

The Barzilai-Borwein is not a monotonous method, but presents second-order convergence behavior, as the expressions of its gradient steps can also be obtained via a quasi-Newton method, more precisely a BFGS method (see [NW06, Chapter 6] for more details).
6 Illustration: Identification of coefficients in an elliptic equation

In this section we propose to address the case of coefficient identification in an elliptic partial differential equations.

6.1 Presentation of the problem

We propose to illustrate the relaxation approach by considering the elliptic equation identification problem in a smooth domain $\Omega$ of $\mathbb{R}^d$ ($d \geq 1$). Denote by $L^\infty_F(\Omega)$ any finite-dimensional subspace of $L^\infty(\Omega)$, with $\dim(L^\infty_F(\Omega)) = n$. For instance, $L^\infty_F(\Omega)$ can be made of piecewise constant functions on a given grid of $\Omega$. Define for $M > m > 0$ the following set:

$$U_{m,M} = \{ u \in L^\infty_F(\Omega) | m \leq u(x) \leq M, \text{ for a.e. } x \in \Omega \}.$$

The control-to-state mapping is given by $S : \mathcal{U} := U_{m,M} \ni u \mapsto y \in \mathcal{Y} := H^1(\Omega)$, where $y$ satisfies the following system in $Q' = H^{-1}(\Omega)$, with $Q := H^1_0(\Omega)$:

$$\begin{cases}
-\text{div}(u \nabla y) = f, & \text{in } \Omega, \\
y|_{\partial\Omega} = g, & \text{on } \partial\Omega.
\end{cases} \tag{21}$$

In system (21) the right-hand-sides $f$ and $g$ are given. Standard results related to system (21) are presented in Appendix A.2.

On the inner-problem

Given $w \in \mathcal{W}$, as inner-problem we consider the following one:

$$\begin{align*}
\min_{(y,u) \in \mathcal{Y} \times \mathcal{U}} & \quad \frac{1}{2} \|y - \hat{z}\|_{L^2(\Omega)}^2 + r(w, u), \\
\text{subject to} & \quad (21).
\end{align*} \tag{\star}$$

Note that we have chosen $\gamma \equiv \text{Id}$, and thus necessarily $n_{\text{out}} = 1$. The gradient for problem (\star) is given by

$$G(u, w, \hat{z}) = r'(u, w) + \int_\Omega \hat{u} \nabla y \cdot \nabla p \, d\Omega.$$

Following Proposition 5, the adjoint state $p \in Q = H^1_0(\Omega)$ is given as the solution of system (5), which corresponds to the following one:

$$\begin{cases}
-\text{div}(u \nabla p) = -(y - y_d) & \text{in } \Omega, \\
p|_{\partial\Omega} = 0 & \text{on } \partial\Omega.
\end{cases} \tag{22}$$

Proving existence and uniqueness of solutions for system (22) is straightforward.

On the relaxed outer-problem

The relaxed outer-problem is formulated as follows:

$$\begin{align*}
\min_{w \in \mathcal{W}} & \quad \frac{1}{2K} \sum_{k=1}^K \|u_k - \hat{u}_k\|_{L^\infty_F(\Omega)}^2 + \frac{\nu}{2} \|w\|_{W^1}^2, \\
\text{subject to:} & \quad G(u_k, w, \hat{z}_k) = 0 \text{ for all } 1 \leq k \leq K.
\end{align*} \tag{\star\star}$$

Instead of detailing the different steps of the theoretical findings for solving (\star\star) in this particular example, we leave this task to the reader, and prefer to focus now on the numerical results.

In what follows, we consider $\Omega = (0, 1)$ (and thus $d = 1$), and discretize the different elliptic equations with P1-finite elements on a uniform mesh made of 100 subdivisions.
6.2 Rediscovering the $L^2$-norm, or not

Data $\{(u_k, y_k); 1 \leq k \leq K\}$ are generated artificially by solving the inner-problems corresponding to the $L^2$ regularizer $u \mapsto \frac{1}{2} \|u\|_U^2$, with $U = \mathbb{R}$. Next a feed-forward neural network is trained with this data set. The purpose of this test is to compare the trained neural network with the $L^2$ function that represents the ground-truth regularizer to (re-)discover. Convergence of the objective function is presented in Figure 1, with values given in Table 1. Figure 2 represents the graph of the trained ANN compared with the one of the regularizer with which we generated the artificial data, namely $u \mapsto \frac{1}{2} |u|_U^2$.

![Misfit evolution during gradient steps for training a neural network from data generated with a $L^2$-regularizer, in linear scale (left) and log scale (right).](image1)

![Graph of the trained neural network in blue, and of the ground-truth $L^2$ regularizer in red.](image2)
Table 1: Misfit values (in %) through the different gradient steps.

| steps | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 | 11 | 12 | 13 | 14 | 15 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| misfit (%) | 54.6 | 31.1 | 18.4 | 5.62 | 2.00 | 1.25 | 1.14 | 7.09 | 10.2 | 2.83 | 2.00 | 1.17 | 1.16 | 1.15 | 1.14 |
| steps | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| misfit (%) | 1.46 | 1.19 | 1.18 | 1.19 | 2.68 | 1.09 | 0.33 | 0.30 | 0.29 | 0.30 | 0.32 | 0.27 | 0.27 |

In Figure 1 and Table 1 we see that the relative misfit decreased from 54% to 0.27% after a few iterations. The curves of Figure 2 suggest non-uniqueness for Problem (*), as the residual misfit is of the same range as the approximation error made when discretizing (21).

6.3 Regularizers for compensating noise on the data

Given a set of $K = 10$ piecewise constant coefficients $u_{true} = (u_{true,k})_{1 \leq k \leq K}$, we compute the corresponding states $(y_{true,k})_{1 \leq k \leq K}$ solutions of (21). We then introduce artificially a noise following a Gaussian distribution that transforms each $y_{true,k}$ into $y_{noisy,k}$. The pairs $\{(\hat{z}_k, \hat{u}_k) := (y_{noisy,k}, u_{true,k}), 1 \leq k \leq K\}$ constitute our data set on which we will train the neural network as regularizer for the inner-problem. The aim of such a regularizer consists in compensating the effects of the noise. Problem (*) equipped with the so-trained regularizer will then enable us to determine the desired coefficients $u$, in spite of the noisy state $\hat{z}$ on which we can rely only.

![Misfit evolution during gradient steps for training a noise-compensating neural network, in linear scale (left) and log scale (right).](image-url)
leading to the search of a saddle-point of a Lagrangian functional, given as:

\[ L(y, u, p) = c(y, \hat{z}) + \gamma \circ r(w, u) + \langle p, \varphi(y, u) \rangle_{Q', Q}. \]

Table 2: Misfit values (in %) through the different gradient steps.

| steps | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| misfit (%) | 54.2 | 49.4 | 31.1 | 50.0 | 52.8 | 5.37 | 13.9 | 4.25 | 4.24 | 3.58 | 3.45 | 4.63 | 3.74 | 3.33 | 3.67 | 3.60 | 3.65 | 3.78 |

| steps | 19  | 20  | 21  | 22  | 23  | 24  | 25  | 26  | 27  | 28  | 29  | 30  | 31  | 32  | 33  | 34  | 35  | 36  |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| misfit (%) | 3.19 | 3.50 | 6.60 | 3.70 | 3.90 | 3.30 | 4.19 | 4.41 | 3.19 | 3.50 | 3.19 | 3.91 | 3.91 | 3.75 | 3.19 | 3.91 | 3.94 | 3.79 |

In the illustration presented in Figure 3 and Table 2, the initial misfit – corresponding to a neural network close to zero – is about 54%. We see that few iterations are required for obtaining a misfit lower than 5%. The type of neural network we consider does not allow us to reduce the misfit under 3.32%.

7 Conclusion

We have proposed a method for addressing in practice the data-driven design of regularizers for improving a class of inverse problems formulated as optimal control problems. The class of regularizers is restricted to a set of artificial neural networks, which goes in the sense of the purely nonlinear nature of such a problem. Our approach is bi-level, and based on a relaxation exploiting the necessary optimality conditions for these problems. We saw that such a type of approach enables practical and efficient implementation, providing that the different objective functions and activation functions of the neural network are smooth. Developing a similar approach when quantities involved are less smooth could present an interesting challenge. Further, investigating applicability of our approach for more complex problems, like image segmentation for instance, would demonstrate even more its relevance. In such a context, one may need to take into account neural networks architectures that allow more sophisticated realizations. Finally, noticing that our approach necessitates the evaluation of second-order derivatives of the control-to-state mapping for the inner-problem, developing higher order adjoint-based methods for deriving the optimality conditions for the outer-problem could potentially improve the implementation and the applicability to large-scale problems.

A Appendix

A.1 On the optimal control approach for solving an inverse problem

Let us fix \( w \) for this subsection. A classical approach for solving inverse problems consists in taking into account the equality constraint \( \varphi(y, u) = 0 \) by incorporating it in the functional to minimize, as follows:

\[
\min_{u \in U(y, \varphi) \in \mathbb{Y}} \left( J(y, u, w) := \alpha \| \varphi(y, u) \|_{Q}^2 + c(y, \hat{z}) + \gamma \circ r(w, u) \right),
\]

where \( \alpha > 0 \) is a given parameter and \( w \) is also given. For example, consider the problem of identifying a conductivity represented by the variable \( u \) in the following elliptic partial differential equation:

\[
\begin{aligned}
- \text{div}(u \nabla y) &= f, & \text{in } \Omega, \\
\gamma |_{\partial \Omega} &= g, & \text{on } \partial \Omega.
\end{aligned}
\]

Here we set \( \mathbb{Y} = H^1(\Omega) \), \( Q = H_0^1(\Omega) \) and \( Q' = H^{-1}(\Omega) \). Right-hand-sides are given by \( f \in H^{-1}(\Omega) \) and \( g \in H^{1/2}(\partial \Omega) \). In that case, the contribution of the state equation in the functional to minimize is \( \| \text{div}(u \nabla y) - f \|_{Q}^2 \). This approach would imply differentiation of the data, and thus could lead to approximation biases, especially when the set of data is incomplete. Instead of that, we consider the state equation as a constraint, that we impose by introducing a Lagrange multiplier denoted by \( p \in Q \simeq Q' \), namely the costate or adjoint variable, leading to the search of a saddle-point of a Lagrangian functional, given as:

\[
\mathcal{L}(y, u, p) = c(y, \hat{z}) + \gamma \circ r(w, u) + \langle p, \varphi(y, u) \rangle_{Q, Q'}.
\]
A saddle-point of \( \mathcal{L} \) is obtained as a critical point with respect to the variables \((y, u)\) and \(p\). The duality product \( \langle p, \varphi(y, u) \rangle_{Q, Q'} \) corresponds to a weak variational formulation, namely
\[
\langle p, \varphi(y, u) \rangle_{Q, Q'} = \int_\Omega p \cdot \text{div}(u \nabla y) \, d\Omega - \int_\Omega p \cdot f \, d\Omega
\]
\[
= -\int_\Omega \nabla p : (u \nabla y) \, d\Omega + \int_{\partial \Omega} p \cdot \left( \frac{\partial y}{\partial n} \right) \, d\partial\Omega - \int_\Omega \nabla p : (u \nabla y) \, d\Omega,
\]
and thus we see that the data represented by the control function \(u\) does not need to be differentiated.

### A.2 Results related to the inverse conductivity problem

Recall that \(H^{-1}(\Omega) = (H_0^1(\Omega))'\). It is well-known that for all \(u \in U_{m,M}, f \in H^{-1}(\Omega)\) and \(g \in H^{1/2}(\partial \Omega)\) system (21) admits a unique solution \(y \in H^2(\Omega)\). Moreover, there exists a constant \(C > 0\) depending only on \(\Omega\) such that the following estimate holds:
\[
\|y\|_{H^1(\Omega)} \leq \frac{1}{m} \left( \|f\|_{H^{-1}(\Omega)} + M \|g\|_{H^{1/2}(\partial \Omega)} \right).
\]  

Furthermore, the following result show that the control-to-state mapping \(S\) is Lipschitz:

**Lemma 2.** Let be \(M > m > 0\) and assume that \(u_1, u_2 \in U_{m,M}\). Then the solutions \(y_1\) and \(y_2\) of system (21) corresponding to \(u = u_1\) and \(u = u_2\) respectively satisfy
\[
\|y_1 - y_2\|_{H^1(\Omega)} \leq \frac{1}{m^2} \|u_1 - u_2\|_{L^\infty(\Omega)} \left( \|f\|_{H^{-1}(\Omega)} + M \|g\|_{H^{1/2}(\partial \Omega)} \right),
\]
where \(\|\varphi\|_{H^1(\Omega)} := \|\nabla \varphi\|_{L^2(\Omega)}\).

**Proof.** Substituting the first equations of (21) satisfied by \(y_1\) and \(y_2\) leads us to
\[
-\text{div}(u_1 \nabla y_1) + \text{div}(u_2 \nabla y_2) = 0 \Rightarrow -\text{div}(u_1 \nabla (y_1 - y_2)) + \text{div}((u_2 - u_1) \nabla y_2) = 0.
\]

Note that \(y_1 - y_2 \in H^1_0(\Omega)\). Then, taking the inner product of the last identity above by \(y_1 - y_2\) and integrating by parts yields
\[
\int_\Omega u_1 |y_1 - y_2|^2 \, d\Omega = \int_\Omega (u_2 - u_1) \nabla y_2 \cdot \nabla (y_1 - y_2) \, d\Omega
\]
\[
\Rightarrow m \|\nabla (y_1 - y_2)\|_{L^2(\Omega)}^2 \leq \|u_1 - u_2\|_{L^\infty(\Omega)} \|\nabla y_2\|_{L^2(\Omega)} \|\nabla (y_1 - y_2)\|_{L^2(\Omega)}
\]
\[
\Rightarrow m \|y_1 - y_2\|_{H^1_0(\Omega)} \leq \|u_1 - u_2\|_{L^\infty(\Omega)} \|y_2\|_{H^1_0(\Omega)}.
\]

Using estimate (23) satisfied by \(y_2\), the previous one implies
\[
m \|y_1 - y_2\|_{H^1_0(\Omega)} \leq \|u_1 - u_2\|_{L^\infty(\Omega)} \frac{1}{m} \left( \|f\|_{H^{-1}(\Omega)} + M \|g\|_{H^{1/2}(\partial \Omega)} \right).
\]

Thus estimate (24) follows.

**On the inner-problem**

Let be \(w \in W\). Considering \(y_d \in L^2(\Omega)\), the inner-problem is formulated as follows:
\[
\min_{(y,u)\in H^1(\Omega)\times U_{m,M}} J(y, u, w) = \frac{1}{2} \|y - y_d\|_{L^2(\Omega)}^2 + \gamma \circ r(w, u)
\]
subject to (21).

The result below establishes existence of minimizers for Problem (25).

**Lemma 3.** Given \(w \in W\), Problem (25) admits global minimizers.
Proof. Since the function $J$ of Problem (25) is bounded from below, and since system (21) is well-posed, $J$ admits a minimizing sequence $(y_n, u_n)$ of feasible solutions. From the definition of $U_{m,M}$, the sequence $(u_n)_n$ is uniformly bounded in $L^\infty_0(\Omega)$, which is of finite dimension. Therefore, up to extraction, the sequence $(u_n)_n$ converges strongly in $L^\infty(\Omega)$ towards $u \in U_{m,M}$. From estimate (23), the sequence $(y_n)_n$ is bounded in $H^1(\Omega)$. In virtue of the Banach-Alaoglu theorem, up to extraction it converges weakly towards $y \in H^1(\Omega)$. Let us show that $y$ is a solution of system (21) corresponding to $u$. For all $\varphi \in H^1_0(\Omega)$, using the Green formula, we have

$$
\langle -\text{div}(u \nabla y) - f; \varphi \rangle_{H^{-1}(\Omega), H^1_0(\Omega)} = \langle -\text{div}(u \nabla y) + \text{div}(u_n \nabla y_n); \varphi \rangle_{H^{-1}(\Omega), H^1_0(\Omega)}
$$

$$
= \langle u \nabla y - u_n \nabla y_n; \nabla \varphi \rangle_{L^2(\Omega)} + \langle u \nabla (y - y_n); \nabla \varphi \rangle_{L^2(\Omega)}
$$

$$
= \langle (u - u_n) \nabla y_n; \nabla \varphi \rangle_{L^2(\Omega)} + \langle \text{div}(u \nabla \varphi) - y - y_n; \nabla \varphi \rangle_{L^2(\Omega)}
$$

where we have used that $y - y_n \in H^1_0(\Omega)$. This yields

$$
\left| \langle -\text{div}(u \nabla y) - f; \varphi \rangle_{H^{-1}(\Omega), H^1_0(\Omega)} \right| \leq \|u - u_n\|_{L^\infty(\Omega)} \|\nabla y_n\|_{L^2(\Omega)} + \|\text{div}(u \nabla \varphi) - y - y_n\|_{H^{-1}(\Omega), H^1_0(\Omega)}
$$

Passing to the limit, we deduce that $-\text{div}(u \nabla y) = f$ in $H^{-1}(\Omega)$, and thus that $(y, u)$ is a solution of Problem (25).

Recall finally the following Lipschitz estimate for the Calderon problem (see for instance [Uhl14, Theorem 2.6], referring to [Ale88]):

**Lemma 4.** Assume that $\Omega$ is a bounded open set with smooth boundary, with $d \geq 3$. For $j = 1, 2$ and $s > d/2$, denote by $u_j \in H^{s+2}(\Omega) \cap A_{m,M}$ satisfying

$$
\|u_j\|_{H^{s+2}(\Omega)} \leq M.
$$

There exists a constant $C = C(\Omega, d, M, m, s) > 0$ such that

$$
\|u_1 - u_2\|_{L^\infty(\Omega)} \leq C \omega \left( \|u_1 - u_2\|_{L^2(H^{1/2}(\partial \Omega), H^{-1/2}(\partial \Omega))} \right),
$$

where $\omega : x \mapsto |\log(x)|^{-\theta} + x$, for some $0 < \theta < 1$.

**References**

[Ale88] Giovanni Alessandrini. Stable determination of conductivity by boundary measurements. *Appl. Anal.*, 27(1-3):153–172, 1988.

[AO17] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124012, 28, 2018.

[GPEB19] Philipp Grohs, Dmytro Perekrestenko, Dennis Elbrächter, and Helmut Bölcskei. Deep neural network approximation theory. *ArXiv*, abs/1901.02220, 2019.

[HKB18] Gernot Holler, Karl Kunisch, and Richard C. Barnard. A bilevel approach for parameter learning in inverse problems. *Inverse Problems*, 34(11):115012, 28, 2018.

[HN20] M. Haltmeier and Linh V. Nguyen. Regularization of inverse problems by neural networks. *ArXiv*, abs/2006.03972, 2020.

[HPUU09] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*, volume 23 of Mathematical Modelling: Theory and Applications. Springer, New York, 2009.

[KP13] Karl Kunisch and Thomas Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM J. Imaging Sci.*, 6(2):938–983, 2013.
[LOS18] Sebastian Lunz, Ozan Öktem, and Carola-Bibiane Schönlieb. Adversarial regularizers in inverse problems. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, Advances in Neural Information Processing Systems 31, pages 8507–8516. Curran Associates, Inc., 2018.

[LSAH20a] Housen Li, Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. NETT: solving inverse problems with deep neural networks. Inverse Problems, 36(6):065005, 23, 2020.

[LSAH20b] Housen Li, Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. NETT: solving inverse problems with deep neural networks. Inverse Problems, 36(6):065005, jun 2020.

[MDS+20] Subhadip Mukherjee, Sören Dittmer, Zakhar Shumaylov, Sebastian Lunz, Ozan Öktem, and Carola-Bibiane Schönlieb. Learned convex regularizers for inverse problems, 2020.

[Nes83] Yu. E. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. Dokl. Akad. Nauk SSSR, 269(3):543–547, 1983.

[NW06] Jorge Nocedal and Stephen J. Wright. Numerical optimization. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[ONSH19] Daniel Obmann, Linh Nguyen, Johannes Schwab, and Markus Haltmeier. Augmented nett regularization of inverse problems, 2019.

[ONSH20] D. Obmann, L. Nguyen, J. Schwab, and M. Haltmeier. Sparse anett for solving inverse problems with deep learning. In 2020 IEEE 17th International Symposium on Biomedical Imaging Workshops (ISBI Workshops), pages 1–4, 2020.

[OSH19] D. Obmann, J. Schwab, and M. Haltmeier. Sparse synthesis regularization with deep neural networks. In 2019 13th International conference on Sampling Theory and Applications (SampTA), pages 1–5, 2019.

[Ray97] M. Raydan. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J. Optim., 7(1):26–33, 1997.

[Uhl14] Gunther Uhlmann. 30 years of Calderón’s problem. In Séminaire Laurent Schwartz—Équations aux dérivées partielles et applications. Année 2012–2013, Sémin. Équ. Dériv. Partielles, pages Exp. No. XIII, 25. École Polytech., Palaiseau, 2014.