Journal of Machine Learning Research 24 (2025) 1-41 Submitted 10/24; Revised 7/25; Published 10/25

Jackpot: Approximating Uncertainty Domains with Adversarial Manifolds

Nathanaël Munier NATHANAEL.MUNIER@ENS-RENNES.FR
University of Toulouse, CNRS, IMT, CBI, France

Emmanuel Soubies
University of Toulouse, CNRS, IRIT, CBI, France

Pierre Weiss
University of Toulouse, CNRS, IRIT, CBI, France

Editor: Joseph Salmon

Contents

1 Introduction **2**
1.1 Application Examples . . . 3
1.1.1 Identifying Dynamical Systems Parameters . . . 3
1.1.2 Blind Inverse Problems . . . 4
1.1.3 Bayesian Posterior Exploration . . . 4
1.2 Related Works . . . 5
1.2.1 Structural Identifiability . . . 5
1.2.2 Practical Identifiability . . . 6
1.3 Main Contribution . . . 8

2 The Jackpot Algorithm and its Guarantees **8**
2.1 Linear Approximation . . . 9
2.1.1 Approximation Result . . . 9
2.1.2 Lowest Singular Vectors Computation . . . 10
2.2 Nonlinear Approximation . . . 11
2.2.1 The Jackpot Manifold . . . 11
2.2.2 Approximation Guarantees . . . 13
2.2.3 Numerical Computation . . . 15
2.2.4 Discussion . . . 17

3 Numerical Experiments **20**
3.1 Measuring Masses in the Solar System . . . 20
3.2 Blind Deblurring . . . 23
3.3 Posterior Exploration for an Image Deblurring Problem . . . 24

A Proofs **28**
A.1 Definition and Characterization of Manifolds . . . 28

A.2 Linear Approximation 29
A.2.1 Proof of Proposition 4 29
A.3 Nonlinear Approximation 30
A.3.1 Proof of Theorem 5 31
A.3.2 Proof of Theorem 8, Theorem 9, Proposition 11 and Theorem 12 . . 33
A.4 Numerical Computation 36
A.4.1 Proof of Proposition 14 36
A.4.2 Proof of Corollary 15 37

Abstract

Given a forward mapping $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ and a point $x^\star \in \mathbb{R}^N$, the region $\{x \in \mathbb{R}^N, \|\Phi(x) - \Phi(x^\star)\|_2 \leq \varepsilon\}$, where $\varepsilon \geq 0$ is a perturbation amplitude, represents the set of all possible inputs x that could have produced the measurement $\Phi(x^\star)$ within an acceptable error margin. This set is related to uncertainty analysis, a key challenge in inverse problems. In this work, we develop a numerical algorithm called Jackpot (Jacobian Kernel Projection Optimization) which approximates this set with a low-dimensional *adversarial manifold*. The proposed algorithm leverages automatic differentation, allowing it to handle complex, high dimensional mappings such as those found when dealing with dynamical systems or neural networks. We demonstrate the effectiveness of our algorithm on various challenging large-scale, non-linear problems including parameter identification in dynamical systems and blind image deblurring.

Keywords: Inverse problem, uncertainty quantification, identifiability

1. Introduction

Let $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ denote an arbitrary C^1 mapping and $y = \Phi(x) + \eta$ be some noisy measurement of $x \in \mathbb{R}^N$ through this mapping. Given an estimate $x^\star$ of x, our main objective in this paper is to design a numerical algorithm to describe the set $\mathcal{U}^\varepsilon$ defined by

$$\begin{aligned} \mathcal{U}^\varepsilon &\stackrel{\text{def}}{=} \left\{x \in \mathbb{R}^N, \|\Phi(x) - \Phi(x^\star)\|_2 \leq \varepsilon\right\} \\ &= \Phi^{-1}\left(\mathcal{B}\left(\Phi(x^\star), \varepsilon\right)\right), \end{aligned} \tag{1}$$

for some $\varepsilon \geq 0$, where $\|\cdot\|_2$ denotes the standard ℓ^2-norm and $\mathcal{B}(y, r)$ denotes an ℓ^2-ball centered at y of radius r. For example, choosing $\varepsilon = \|\Phi(x^\star) - y\|$ defines $\mathcal{U}^\varepsilon$ as the set of points whose forward images under Φ are as close to the measurement y as that of $x^\star$, and can thus be interpreted as equally plausible estimates. As will be explained later, many important practical problems can be framed in this manner.

If the mapping Φ is linear, then $\mathcal{U}^\varepsilon$ is an ellipsoid that can be characterized using linear algebra tools. In the general nonlinear case, the set $\mathcal{U}^\varepsilon$ is the sublevel set of a near arbitrary function. Indeed, an arbitrary closed set $\mathcal{E}$ can be written as the 0-level set of the function $\Phi(x) = \text{dist}(x, \mathcal{E})^2$. Given the level of generality, it is unreasonable to expect solving the problem properly for all mappings Φ. Describing $\mathcal{U}^\varepsilon$ is more challenging than finding all the global minimizers of a near arbitrary non convex function.

Therefore, we aim for a more modest goal: describe $\mathcal{U}^\varepsilon$ locally around the point $x^\star$. Additionally, we seek to approximate this set by another one of relatively low intrinsic dimensionality. This serves two purposes: i) to design tractable algorithms and ii) to provide an output that can be easily visualized. Our main objective in this paper can therefore be summarized as follows.

Main objective Given a mapping $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ and a point $x^\star \in \mathbb{R}^N$, find a D-dimensional manifold[1] $\mathcal{M}_\delta^\varepsilon$ that approximates the set

$$\mathcal{U}_\delta^\varepsilon \stackrel{\text{def}}{=} \{x \in \mathcal{B}(x^\star, \delta),\ \|\Phi(x) - \Phi(x^\star)\|_2 \leq \varepsilon\}, \tag{2}$$

where $\delta \geq 0$ bounds the diameter of the approximating manifold. The set $\mathcal{U}_\delta^\varepsilon$ can be encountered under different names, depending on the scientific field: *equivalence domain* (Grayver and Kuvshinov, 2016), *low misfit region* (Fernández-Martínez et al., 2012), *solution set* (Jauberthie et al., 2013) or *uncertainty region* (Fernández-Martínez and Fernández-Muñiz, 2020). Throughout this paper, we will call it *uncertainty region*.

We are particularly interested in complex models Φ (e.g. dynamical systems, neural networks), which can lead to problems involving high-dimensional (latent) spaces. Throughout the paper, we assume that the Jacobian of Φ can be accessed either through analytical derivation or by using automatic differentation algorithms.

1.1 Application Examples

Many problems involve identifying parameters of physical, biomedical, or chemical systems from indirect observations. The primary applications we have in mind are related to the field of inverse problems. In this field, the set $\mathcal{U}^\varepsilon$ describes what can be identified in a system and what cannot. Let us illustrate this point with a few examples studied in the numerical experiments.

1.1.1 Identifying Dynamical Systems Parameters

Assume that some dynamical system $u : \mathbb{R}_+ \to \mathbb{R}^{P\times D}$ of P particles in $\mathbb{R}^D$ is governed by a first or second order equation:

$$\dot{u}(t) = f(u(t), x) \qquad \text{or} \qquad \ddot{u}(t) = f(u(t), x)$$

where $u(t) = (u_1(t), \ldots, u_P(t))$ denotes the particles positions and $f : \mathbb{R}^{P\times D} \times \mathbb{R}^N \to \mathbb{R}^{P\times D}$ is a force or velocity field parameterized by the vector $x \in \mathbb{R}^N$. After measuring the particles positions at multiple time points, we would like to recover the unknown parameter x that describes the dynamical system.

An example that we will be studied later is the solar system, where $u(t)$ represents the positions of $P = N$ planets at time t, f are forces given by Newton's second law of attraction and $x \in \mathbb{R}^N$ is the mass of the different planets. We assume that a measurement system (e.g. telescope) returns estimates of the positions u at some times, leading to a measurement vector $(y_1, \ldots, y_K)$ of the form $y_k = u(t_k) + b_k$ for sampling times $t_1 < \cdots < t_K$ and unknown measurement errors b_k.

The scientific question we aim to address is: *can we recover the mass of the planets from the partial observation of their positions?* To this end, we can set $\Phi : x \mapsto (u(t_1), \ldots, u(t_K))$ and find a first guess $x^\star$ of the mass using a least square regression:

$$x^\star = \operatorname*{argmin}_{x\in\mathbb{R}^N} \frac{1}{2}\|\Phi(x) - y\|_2^2.$$

1. See Appendix A.1 for a definition.

Describing the set $\mathcal{U}_\delta^\varepsilon$ provides a much finer description of the uncertainties than the single estimate $x^\star$. As we will demonstrate, our algorithm reveals that predicting the mass of Pluto is highly challenging. This observation aligns with historical estimates: initially, Pluto's mass was thought to be twice that of Earth, but in 2015, the New Horizons probe measured it to be just 1/455th of Earth's mass.

1.1.2 Blind Inverse Problems

Assume that we acquire measurements y of the form

$$y = \mathcal{P}(A(\theta)(x)),$$

where $\mathcal{P} : \mathbb{R}^M \to \mathbb{R}^M$ represents a perturbation (e.g. noise, quantization) and $\theta \in \mathbb{R}^N$ is a parameter that describes the state of an acquisition system $A(\theta) : \mathbb{R}^{N'} \to \mathbb{R}^M$. This parameter could represent an unknown point spread function in optics or uncertain projection angles in tomography for instance. The goal in blind inverse problems is to recover both x and θ from the measurements y.

One approach we will explore in the numerical section involves constructing a learned reconstruction mapping $R(A(\theta), y)$, which, given a forward operator $A(\theta)$ and a measurement vector y, outputs an estimate $\hat{x}(\theta) = R(A(\theta), y)$ of x. Then, a simple approach to estimate θ is to minimize the discrepancy:

$$\operatorname*{argmin}_{\theta \in \mathbb{R}^N} \frac{1}{2} \left\| A(\theta)(\hat{x}(\theta)) - y \right\|_2^2,$$

ensuring that the pair $(A(\theta), \hat{x}(\theta))$ is coherent with the observation y.

By setting $\Phi(\theta) = A(\theta)(\hat{x}(\theta))$ and exploring $\mathcal{U}_\delta^\varepsilon$ around a local minimizer $\theta^\star$, we can identify all pairs coherent with the data. This is a way to describe some uncertainty properties of the blind inverse problem.

1.1.3 Bayesian Posterior Exploration

Similar to the previous example, assume that a system yields measurements y of the form

$$y = \mathcal{P}(A(x)),$$

where $A : \mathbb{R}^N \to \mathbb{R}^M$ is a known operator describing the acquisition device. Whenever A is not invertible, recovering x from y requires regularization. An elegant approach to formalize this principle uses Bayesian reasoning: we assume that x is the realization of some random vector $\mathbb{x}$ with probability distribution function $p_{\mathbb{x}}$. Under the assumption that the prior $p_{\mathbb{x}}$ is a proper probability distribution and the likelihood $p_{\mathbb{y}|\mathbb{x}}(y|x)$ is well-defined for the observed data y, it is possible to construct a posterior distribution:

$$p_{\mathbb{x}|\mathbb{y}}(x|y) = \frac{p_{\mathbb{y}|\mathbb{x}}(y|x) \cdot p_{\mathbb{x}}(x)}{p_{\mathbb{y}}(y)}$$

and Bayesian estimators such as the Maximum A Posteriori (MAP) estimate:

$$\begin{aligned} \hat{x}_{\mathrm{MAP}}(y) &\overset{\text{def}}{=} \operatorname*{argmax}_{x \in \mathbb{R}^N} p_{\mathbb{x}|\mathbb{y}}(x|y) \\ &= \operatorname*{argmin}_{x \in \mathbb{R}^N} -\log\left(p_{\mathbb{y}|\mathbb{x}}(y|x)\right) - \log\left(p_{\mathbb{x}}(x)\right). \end{aligned}$$

The prior $p_{\mathbb{x}}$ can be handcrafted or learned with large databases. The latter strategy oftentimes provides surprisingly good results, which are rapidly replacing older strategies (Nah et al., 2021; Muckley, 2021; Sidky and Pan, 2024). In scientific applications, it however becomes critical to certify the results, i.e. describe what information can be safely regarded as valid and the one which is uncertain.

One approach to achieving this involves describing the high probability region

$$H^\alpha = \{x \in \mathbb{R}^N,\ \log p_{\mathbb{x}|\mathbb{y}}(x|y) \geq \alpha\} \tag{3}$$

for some probability threshold α. However, the prior $p_{\mathbb{x}}(x)$ can often only be accessed indirectly through its gradient (Hyvärinen, 2005; Vincent, 2011). In that situation, another approach is to compute a local minimizer $x^\star$ of the posterior distribution and to explore the set

$$\mathcal{U}_\delta^\varepsilon = \left\{x \in \mathcal{B}(x^\star, \delta), \|\nabla \log p_{\mathbb{x}|\mathbb{y}}(x|y)\|_2 \leq \varepsilon\right\}. \tag{4}$$

This problem fits our formalism with $\Phi = \nabla \log p_{\mathbb{x}|\mathbb{y}}(\cdot|y)$ and $\Phi(x^\star) = 0$, since $x^\star$ is a minimizer. When $x^\star = \hat{x}_{\text{MAP}}$, we have $\mathcal{U}_\delta^\varepsilon \subseteq H^\alpha$ for some parameter $\varepsilon > 0$ that depends on α and sufficiently small $\delta > 0$. However the reverse inclusion is not true in general.

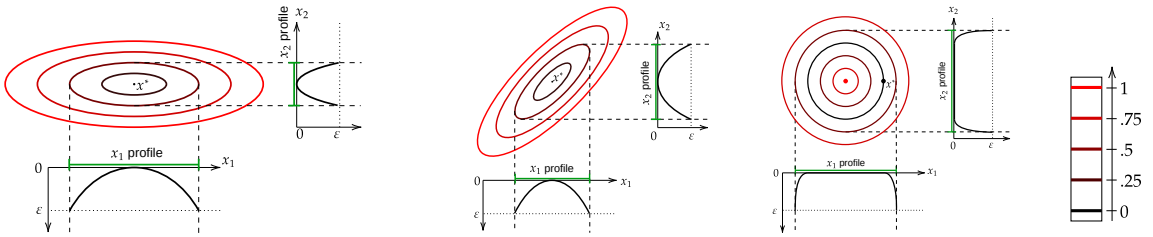
1.2 Related Works

As illustrated by these examples, describing uncertainty regions can be useful in various fields such as system identification, inverse and blind inverse problems, data assimilation. It can also be regarded as a type of sensibility analysis or uncertainty quantification. Let us review some approaches available in the literature. We will use the nomenclature of the identifiability theory (Wieland et al., 2021), which classifies the study as *structural* in the noiseless setting ($\varepsilon = 0$) and *practical* when noise is considered ($\varepsilon > 0$).

1.2.1 Structural Identifiability

Using our notation, *global structural identifiability* means that $\mathcal{U}^0 = \{x^\star\}$ while *local structural identifiability* means that there exists $\delta > 0$ such that $\mathcal{U}_\delta^0 = \{x^\star\}$. The literature devoted to those two notions of *identifiability* is vast. Under mild regularity assumptions on Φ, a necessary and sufficient condition for local structural identifiability is given by $\ker(J_\Phi(x^\star)) = \{0\}$, where $J_\Phi(x^\star) \in \mathbb{R}^{M\times N}$ is the Jacobian matrix of Φ at $x^\star$ (Rothenberg, 1971). When Φ is linear, this condition reads $\ker(\Phi) = \{0\}$ and implies global structural identifiability. Beyond the linear case, there exist rich theories regarding the (local) identifiability of bilinear inverse problems (Li et al., 2016; Kech and Krahmer, 2017), phase retrieval problems (Jaganathan et al., 2016; Grohs et al., 2020), partial differential equations (Villaverde et al., 2018) or neural networks (Phuong and Lampert, 2019; Bona-Pellissier et al., 2022), to name a few.

When $x^\star$ is non-identifiable, the analysis and characterization of the non-trivial set $\mathcal{U}^0$ locally around $x^\star$ has also been the object of many works. Existing methods include computing the uncertainty region $\mathcal{U}^0$ in explicit form using differential algebra (Bates et al., 2019), numerical algebraic geometry (Hubert, 1999; Bellu et al., 2007) or Lie group theory (Merkt et al., 2015). However, these algebraic methods are limited to the analysis of rational functions and become computationally prohibitive as the number of parameters increases.



(a) $\Phi = A$ (b) $\Phi = UAU^T$ (c) $\Phi(x) = \|x\|_2^2$

Figure 1: Illustration of the profile likelihood method. The red-black curves are the level sets of the discrepancy $\|\Phi(\cdot) - \Phi(x^\star)\|_2$. The discrepancy profiles along each axis are plotted in black, together with the confidence intervals in green. In (a), (b), we fixed $A = \mathrm{diag}(.2, 1)$, an unitary matrix U and $x^\star = (0, 0)$. In (c), we fixed $x^\star = (1, 0)$.

1.2.2 Practical Identifiability

Practical identifiability, which guarantees stability to noise on the observations, is the main topic of this paper. Alternative names are sensitivity analysis or uncertainty quantification, which are closely related. For the local characterization of the uncertainty region $\mathcal{U}^\varepsilon$, various approaches exist.

Coordinate-based methods Coordinate-based methods examine the model's sensitivity along each direction of the canonical basis, where the map $x \mapsto \exp\left(-\|\Phi(x) - \Phi(x^\star)\|_2\right)$ is often called likelihood.

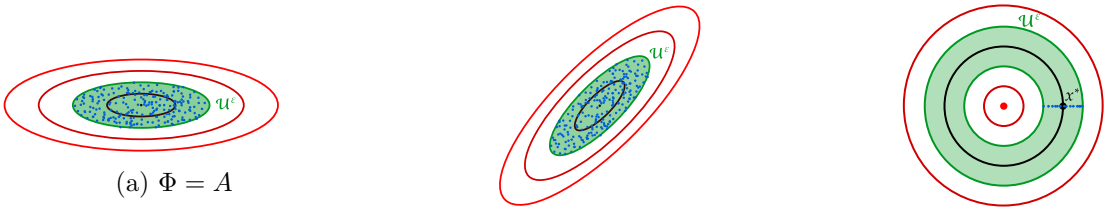
Profile likelihood methods (Venzon and Moolgavkar, 1988; Raue et al., 2009) compute one-dimensional profiles. The profiles are evaluated by fixing one coordinate and optimizing over the remaining ones:

$$\min_{\substack{x \in \mathbb{R}^N \\ x_i = x_i^\star + t_j}} \|\Phi(x) - \Phi(x^\star)\|_2.$$

This is done for different deviations $t_j \in \mathbb{R}$ and all coordinates i. The directions i, where the likelihood is high are uncertain. An illustration with simple likelihood functions is given in Fig. 1. Notice that in practice, only endpoints of the confidence intervals are computed (Fischer and Lewis, 2021). A weakness of these approaches is that they act component-wise. Hence, in Fig. 1 examples (a), (b), we see that the profiles change a lot depending on the coordinate system, showing that the profile directions should be chosen with care. In Fig. 1 (c) for instance, the profiles are locally flat in both directions, while the uncertainty is only one-dimensional along the unit circle.

A possible approach to avoid this drawback, is to find meaningful directions using the Fisher information matrix, defined by $J_\Phi(x^\star)^T \cdot J_\Phi(x^\star)$. The eigenvectors associated to the lowest singular values indicate the directions which are hardest to identify. This idea was used by Eisenberg and Hayashi (2014) to identify combinations of coordinates which are jointly identifiable and draw profiles likelihood for these subsets only. A similar mechanism is at the core of our approach, though used in a different manner.

Monte Carlo sampling methods The problem can be viewed as recovering the pre-image $\Phi^{-1}(\mathcal{B}(\Phi(x^\star), \varepsilon))$, i.e., all points that Φ maps into a ball around $\Phi(x^\star)$. A simple way



(a) $\Phi = A$ (b) $\Phi = UAU^T$ (c) $\Phi(x) = \|x\|_2^2$

Figure 2: Illustration of Monte Carlo sampling on the same examples as in Fig. 1. The set $\mathcal{U}^\varepsilon$ is represented in green and the sampling points are in blue.

to do so is to deploy a Monte Carlo sampling method (Mosegaard and Tarantola, 1995). A few samples y_k for $1 \leq k \leq K$ are drawn at random within $\mathcal{B}(\Phi(x^\star), \varepsilon)$. Then, the antecedents can be computed as

$$\hat{x}_k \stackrel{\text{def}}{=} \underset{x \in \mathbb{R}^N}{\operatorname{argmin}} \|\Phi(x) - y_k\|_2^2. \tag{5}$$

The discrete set $(\hat{x}_k)$ provides a snapshot of $\mathcal{U}^\varepsilon$. An illustration is provided in Fig. 2 with our basic examples.

Statistical analyses of the samples $(\hat{x}_k)$ can then be performed to quantify the uncertainty. These include, for instance, variance or covariance estimates, bootstrapping (Hengl et al., 2007), sensitivity-based methods (Helton et al., 2006; Bardsley, 2012), or randomize-then-optimize strategies (Bardsley et al., 2014).

As opposed to the profile likelihood approach, Monte Carlo sampling accounts for relationships between coordinates. Yet, it suffers from multiple weaknesses.

- It can be computationally expensive given that each sample $\hat{x}_k$ requires solving an optimization problem and that the number of sampling points required for statistical analyses grows exponentially with the dimension of the problem (Fernández-Martínez and Fernández-Muñiz, 2020; Fernández-Martínez et al., 2013).

- For degenerate problems, where Φ is non injective, the optimization problem may possess multiple minimizers. In those situations, sampling becomes inefficient. In Fig. 2 (c) for instance, we started the optimization algorithm by setting $x_{\text{init}} = x^\star$ as an initialization point. As a result, the sampling approach only yields a 1D subset of $\mathcal{U}^\varepsilon$, that does not capture the complete annulus.

- Uniform sampling in $\mathcal{B}(\Phi(x^\star), \varepsilon)$ might not translate to a uniform sampling in $\mathcal{U}^\varepsilon$, potentially over-representing some regions and under-representing others. This issue will be striking in our numerical experiments.

Some of these challenges can be addressed with advanced diffusion models, which have recently gained significant attention in artificial intelligence research. These methods were initially developed for generative modeling tasks (Ho et al., 2020) and have since been adapted for various inverse problems (Chung et al., 2022; Laumont et al., 2022; Chemseddine et al., 2024). Ongoing research focuses on certifying the accuracy of posterior sampling (Thong et al., 2024) and improving the efficiency of sample generation. However, given the computational cost of these approaches, we have chosen not to use them for comparisons.

Geometric analysis Finally, the last class of methods, the one we endorse, aims to compute geometrically the structure of the set $\mathcal{U}^\varepsilon$. Arutjunjan et al. (2022) proved that the boundary of $\mathcal{U}^\varepsilon$ is a $N-1$-dimensional manifold and proposed a numerical method to compute its boundary. It relies on the construction of a complete vector field tangential to the level sets of $\|\Phi(x) - \Phi(x^\star)\|_2$. Then, the determination of the boundary of $\mathcal{U}^\varepsilon$ amounts to solve an ODE. This approach is limited to low-dimensional problems such as the two-dimensional non-polynomial and was applied only to three-dimensional polynomial by Arutjunjan et al. (2022). Instead of considering the whole space, Raman et al. (2017) proposed to determine a trajectory referred to as *minimally disruptive curve* or *talweg* within the structural or practical non-identifiable sets $\mathcal{U}^0$ and $\mathcal{U}^\varepsilon$. This trajectory is a one-dimensional sub-manifold approximation of $\mathcal{U}^\varepsilon$ with the smallest discrepancy values. However, it is unclear how to extend this approach to higher dimensions. Another approach given by Transtrum and Qiu (2014) consists in successively eliminating one parameter at a time to finally obtain a partial low-dimensional boundary of the uncertainty region. This method is not applicable to high-dimensional problems.

For further details and references related to these approaches, we refer the reader to the comprehensive reviews (Miao et al., 2011; Wieland et al., 2021; Lam et al., 2022).

1.3 Main Contribution

We aim at advancing the field of local practical identifiability by designing the Jackpot algorithm. It is applicable to large dimensional problems and yields a manifold of arbitrary (but preferably small) dimension D. This manifold is constructed so as to provide a good approximation of the ε-uncertainty region in Hausdorff distance, regardless of the non-linear and non-algebraic nature of the model. A simple illustration of its output is given in Fig. 5.

The proposed algorithm is based on automatic differentiation, allowing us to tackle high-dimensional problems. It relies on the Python package deepinv (Tachella et al., 2025). In particular, we illustrate its behavior in problems involving dynamical systems and neural networks. The numerical experiments supporting these results, together with the implementation of the *Jackpot* algorithm, are available in the corresponding GitHub repository.

2. The Jackpot Algorithm and its Guarantees

Our main goal is to describe $\mathcal{U}_\delta^\varepsilon = \{x \in \mathcal{B}(x^\star, \delta),\ \|\Phi(x) - \Phi(x^\star)\|_2 \leq \varepsilon\}$, the local $\varepsilon^2/2$-sublevel set of the function

$$F(x) \stackrel{\text{def}}{=} \frac{1}{2}\|\Phi(x) - \Phi(x^\star)\|_2^2. \tag{6}$$

In what follows, to measure how well we approximate the set $\mathcal{U}_\delta^\varepsilon$, we will use Kolmogorov and Hausdorff distances which we define next.

Definition 1 (D-width or Kolmogorov distance) *Let $\mathcal{G}(D, N)$ denote the set of D-dimensional vector subspaces in dimension N (the Grassmanian). The Kolmogorov distance δ_D of a set $\mathcal{S} \subset \mathbb{R}^N$ is defined by:*

$$\delta_D(\mathcal{S}) \stackrel{\text{def}}{=} \inf_{\mathcal{V} \in \mathcal{G}(D,N)} \sup_{x \in \mathcal{S}} \inf_{v \in \mathcal{V}} \|x - v\|_2. \tag{7}$$

It measures how well a set $\mathcal{S}$ can be approximated by a D-dimensional subspace. Let us also recall the notion of Hausdorff distance.

Definition 2 (Hausdorff distance) *For two subsets $\mathcal{X}, \mathcal{Y}$ of a metric space $(\mathcal{M}, d)$, the Hausdorff distance is defined as*

$$d_{\mathcal{H}}(\mathcal{X}, \mathcal{Y}) \stackrel{\text{def}}{=} \max \left\{ \sup_{x \in \mathcal{X}} d(x, \mathcal{Y}), \ \sup_{y \in \mathcal{Y}} d(\mathcal{X}, y) \right\} \tag{8}$$

$$= \inf_{\eta \geq 0} \left\{ \mathcal{X} \subseteq \mathcal{Y} + \mathcal{B}(0, \eta) \text{ and } \mathcal{Y} \subseteq \mathcal{X} + \mathcal{B}(0, \eta) \right\}. \tag{9}$$

This distance intuitively measures how much a set should be dilated to include the other one and vice versa.

In this section, we start by proposing a linear manifold approximation $\widetilde{\mathcal{M}}_\delta^\varepsilon$ obtained from a quadratic approximation of F at $x^\star$. This allows us to present the main approximation results in a simplified format. We then turn to the construction of a more precise nonlinear manifold approximation $\mathcal{M}_\delta^\varepsilon$. We describe its approximation guarantees, and a practical algorithm to compute it.

2.1 Linear Approximation

Under a $\mathcal{C}^2$ regularity assumption, we can approximate F by a quadratic function. Most terms in the Taylor expansion vanish since $x^\star$ is a minimizer and for any $h \in \mathbb{R}^N$, we obtain:

$$F(x^\star + h) = \frac{1}{2} \|J_\Phi(x^\star) \cdot h\|_2^2 + o(\|h\|_2^2). \tag{10}$$

This motivates us to consider the quadratic approximation of F at $x^\star$:

$$\widetilde{F}(x) \stackrel{\text{def}}{=} \frac{1}{2} \|J_\Phi(x^\star) \cdot (x - x^\star)\|_2^2. \tag{11}$$

The uncertainty region can be described as the $\varepsilon^2/2$-sublevel set of this function, denoted $\widetilde{\mathcal{U}}^\varepsilon = \left\{ x \in \mathbb{R}^N, \ \tilde{F}(x) \leq \varepsilon^2/2 \right\}$. Similarly to the definition of $\mathcal{U}_\delta^\varepsilon$, we also set

$$\widetilde{\mathcal{U}}_\delta^\varepsilon = \left\{ x \in \mathcal{B}(x^\star, \delta), \ \tilde{F}(x) \leq \frac{\varepsilon^2}{2} \right\}. \tag{12}$$

2.1.1 Approximation Result

In the linear case, the best approximation of $\widetilde{\mathcal{U}}^\varepsilon$ by a D-dimensional subspace is explicit using the singular vectors of the Jacobian matrix of Φ. We can indeed decompose the Jacobian as $J_\Phi(x) = U(x)\Sigma(x)V^T(x)$, where $U(x) \in \mathbb{R}^{M \times M}$ and $V(x) \in \mathbb{R}^{N \times N}$ are orthogonal matrices and $\Sigma(x) \in \mathbb{R}^{M \times N}$ is a diagonal rectangular matrix with nonnegative and decaying entries. To simplify the notation, we let $J^\star \stackrel{\text{def}}{=} J_\Phi(x^\star) = U^\star \Sigma^\star V^{\star,T}$, where $U^\star = [u_1^\star, \ldots, u_M^\star]$, $V^\star = [v_1^\star, \ldots, v_N^\star]$, $\operatorname{diag}(\Sigma^\star) = [\sigma_1^\star, \ldots, \sigma_{R^\star}^\star, 0, \ldots, 0]$, and $R^\star$ is the rank of $J^\star$. By convention, we let $\sigma_{R^\star+1}^\star, \sigma_{R^\star+2}^\star, \ldots, \sigma_N^\star = 0$. The following description is standard.

Proposition 3 *(From Johansen, 1977) The sublevel set $\widetilde{\mathcal{U}}^\varepsilon$ is a (degenerate) hyper-ellipsoid with axes aligned with the vectors $(v_n^\star)$. The length of the ellipsoid along the semi-axis $v_n^\star$ is given by $\frac{\varepsilon}{\sigma_n^\star}$ for $n \in \{1, \ldots, R^\star\}$ and it is infinite for the other axes.*

The following result is rather standard in approximation theory, see e.g. (Pinkus, 2012, Thm. 1.3). First the uncertainty region is approximated by an affine subspace $\mathcal{L}$ and then by a troncation of this subset which gives the approximation manifold $\widetilde{\mathcal{M}}_\delta^\varepsilon$ of $\widetilde{\mathcal{U}}_\delta^\varepsilon$. Guarantees on the approximation distances are also provided and the proof is provided in Appendix A.2.1.

Proposition 4 *Firstly, the affine $D-$dimensional affine space with $D \geq \dim\ker J_\Phi(x^\star)$ that best approximates $\widetilde{\mathcal{U}}^\varepsilon$ for the Kolmogorov distance is given by $\mathcal{T}_{x^\star} \stackrel{\text{def}}{=} x^\star + \mathcal{V}_D^\star$ with*

$$\mathcal{V}_D^\star \stackrel{\text{def}}{=} \operatorname{span}\left(v_{N-D+1}^\star, \dots, v_N^\star\right). \tag{13}$$

The Kolmogorov distance is given by $\delta_D(\widetilde{\mathcal{U}}^\varepsilon - x^\star) = \frac{\varepsilon}{\sigma_{N-D}^\star}$.

Secondly, for any $\delta > 0$, we define the set

$$\widetilde{\mathcal{M}}_\delta^\varepsilon \stackrel{\text{def}}{=} \left\{ x \in \mathcal{T}_{x^\star}; \left\|\Sigma^\star V^{\star,T}(x - x^\star)\right\|_2^2 \leq \varepsilon^2 \text{ and } \|x - x^\star\|_2 \leq \delta \right\}. \tag{14}$$

It satisfies the following equality:

$$d_{\mathcal{H}}(\widetilde{\mathcal{M}}_\delta^\varepsilon, \widetilde{\mathcal{U}}_\delta^\varepsilon) = \min\left(\frac{\varepsilon}{\sigma_{N-D}^\star},\ \delta\right). \tag{15}$$

This linearized approximate set $\widetilde{\mathcal{M}}_\delta^\varepsilon$ was already described in Fernández-Martínez and Fernández-Muñiz (2020). The bounds in Proposition 4 are inversely proportional to the $(N-D)$th singular value of the Jacobian of Φ at $x^\star$. As such, the "linearized" uncertainty region $\widetilde{\mathcal{U}}_\delta^\varepsilon$ can be well approximated by a low-dimensional (i.e., D small) manifold $\widetilde{\mathcal{M}}_\delta^\varepsilon$ when the singular spectrum of the Jacobian $J_\Phi(x^\star)$ admits few small singular values. A general discussion related to this is provided in Section 2.2.4 (see also Fig. 6).

2.1.2 Lowest Singular Vectors Computation

The results above show that the singular values and singular vectors of the Jacobian $J^\star = J_\Phi(x^\star)$ play a critical role for the description of the set $\widetilde{\mathcal{U}}^\varepsilon$. To compute them, two operation are required: i) evaluating the Jacobian matrix or matrix-vector products with it and ii) computing the singular value decomposition.

Jacobian computation Computing matrix-vector products with the Jacobian can be achieved efficiently using automatic differentiation algorithms. In Pytorch for instance, this can be achieved through the *torch.jvp* and *torch.vjp* functions. If the dimension N or M is sufficiently small, then the whole Jacobian matrix $J^\star$ can be evaluated and stored. If the dimensions are too large, then automatic differentation algorithms make it possible to compute left and right matrix-vector products, i.e. products of the form $J^\star v$ or $u^T J^\star$ for arbitrary directions $u \in \mathbb{R}^M$ and $v \in \mathbb{R}^N$.

Singular vectors computation The computation of singular pairs of $J^\star$ or the eigenpairs of $J^{\star,T}J^\star$ are two equivalent problems. The latter is known as a symmetric eigenvalue problem (Parlett, 1998). Several algorithms, called eigensolvers (Liu et al., 2020), have been developed for this purpose. If the matrix is small enough, traditional decomposition methods based on the QR algorithm can be used.

In high dimension, we adopted the Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) method (Knyazev, 2001; Duersch et al., 2018). It is *matrix-free*: it only requires matrix-vector products and does not process the entire matrix. The computation is achieved by optimizing the generalized Rayleigh quotient. For $X \in \mathbb{R}^{N\times D}$, this quotient is defined as

$$f(X) = \operatorname{Tr}\left(\left(X^T X\right)^{-1} (J^\star X)^T J^\star X\right). \tag{16}$$

As such, finding the D leftmost singular vectors of $J^\star$ consists in finding the minimizer of

$$\operatorname*{argmin}_{X\in \mathrm{St}(N,D)} f(X),$$

where $\mathrm{St}(N,D) \stackrel{\text{def}}{=} \left\{X \in \mathbb{R}^{N\times D} \;;\; X^T X = \mathrm{Id}_D\right\}$ is the set of $N \times D$ orthonormal matrices also called the Stiefel manifold. The LOBPCG algorithm can be interpreted as a manifold gradient descent (Absil et al., 2008) with a carefully designed step-size based on the previous iterates.

2.2 Nonlinear Approximation

We now turn to the approximation of the sublevel sets of F without resorting to a quadratic approximation: we want to find a D-dimensional manifold $\mathcal{M}_\delta^\varepsilon$ that approximates $\mathcal{U}_\delta^\varepsilon$ well.

2.2.1 The Jackpot Manifold

The linear subspace $\mathcal{T}_{x^\star} = x^\star + \mathcal{V}_D^\star$ can be a poor approximation of $\mathcal{U}^\varepsilon$ when Φ is nonlinear. However, it is a natural tangent space for the set $\mathcal{M}_\delta^\varepsilon$ at $x^\star$. The idea behind our algorithm consists in bending this tangent space, to better fit the uncertainty region $\mathcal{U}_\delta^\varepsilon$. A similar idea was proposed by Rheinboldt (1996) for polynomials. However, the algorithm was limited to three dimensions and could not visualize the uncertainty region, as it was constrained to the case where $\varepsilon = 0$.

In mathematical terms, we want to find a one-to-one map γ that sends points of $\mathcal{T}_{x^\star}$ onto $\mathcal{U}^\varepsilon$. This can be achieved thanks to the following optimization problem:

$$\gamma(z) \stackrel{\text{def}}{=} \operatorname*{argmin\text{-}loc}_{V_D^{\star,T}(x-x^\star)=z} \frac{1}{2}\left\|\Phi(x) - \Phi(x^\star)\right\|_2^2, \tag{$\mathcal{P}_z$}$$

where

$$V_D^\star \stackrel{\text{def}}{=} [v_{N-D+1}^\star, \cdots, v_{N-1}^\star, v_N^\star] \in \mathbb{R}^{N\times D}. \tag{17}$$

The constraint $V_D^{\star,T}(x - x^\star) = z$ means that the projection of $\gamma(z)$ on the tangent space $\mathcal{T}_{x^\star}$ has coordinate z. It ensures the injectivity of the map γ. The argmin-loc notation refers to a local minimizer near $x^\star$. The main tools appearing in this description are depicted in Fig. 3.

A simple illustration To illustrate this principle, we consider the mapping $\Phi(x) = \frac{1}{2}\|x\|_2^2$ and set $x^\star = e_1$, where $(e_1, \ldots, e_N)$ is the canonical basis. It is a basic example from a broader context named algebraic implicit curves parameterization. For this specific function, the set $\mathcal{U}^0$ is a sphere of radius 1 centered at 0. The tangent space is the hyperplane orthogonal to e_1 passing through $x^\star$. Simple computation yields $\gamma(z) = \sqrt{1 - \|z\|_2^2}e_1 + \sum_{n=2}^N z_n e_n$. We see

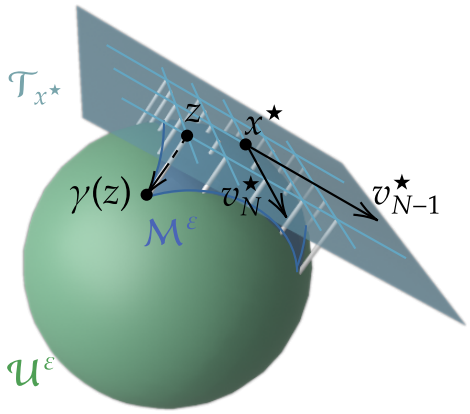


Figure 3: Summary of Jackpot (JACobian Kernel Projection by OpTimization). In this example, we set $\Phi(x) = \|x\|_2^2$, $\varepsilon = 0$ and $x^\star = e_1$. In this setting, the uncertainty region $\mathcal{U}^\varepsilon$ is just a unit sphere, and the affine space $\mathcal{T}_{x^\star}$ is the tangent space to $\mathcal{U}^\varepsilon$ at $x^\star$. The uncertainty region is obtained by projecting the tangent plane onto $\mathcal{U}^\varepsilon$.

that the proposed parameterization allows us to recover the half sphere, which is illustrated in Fig. 3. This example also shows the importance of considering local minimizers, since the problem $(\mathcal{P}_z)$ admits two global minimizers. We provide a few extra toy examples in Fig. 5.

Theorem 5 (Well-definedness of the parameterization γ) *Let $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ be a $\mathcal{C}^1$ map, $x^\star \in \mathbb{R}^N$, and $D \geq \dim \ker J_\Phi(x^\star)$ denote an integer.*

Then there exists a neighborhood $\mathcal{V} \subseteq \mathbb{R}^D$ such that $(\mathcal{P}_z)$ admits a unique solution $\gamma(z)$ for $z \in \mathcal{V}$. Moreover, the mapping $\gamma : \mathcal{V} \to \mathbb{R}^N$ verifies $\gamma(0_D) = x^\star$ and it has the following properties

- *If Φ is of class $\mathcal{C}^2$, then γ is of class $\mathcal{C}^1$.*
- *If Φ is of class $\mathcal{C}^1$ and J_Φ is locally Lipschitz and definable, then γ is a locally Lipschitz definable function.*

Proof The proof is provided in Appendix A.3.1 ■

Remark 6 *The second condition in Theorem 5 is weaker than the first one and allows us to handle a larger class of operators. The notion of definable function (Van den Dries, 1998; Coste, 2000) generalizes the notion of (implicitly defined) semi-algebraic function. It encompasses all functions commonly considered in engineering. We refer the reader to (Bolte et al., 2021, Appendix A.2) for a precise definition.*

Main definition The mapping γ provides a good candidate to approximate the uncertainty region $\mathcal{U}_\delta^\varepsilon$. It writes as follows.

Definition 7 (The Jackpot "manifold") *For $\delta > 0$ sufficiently small, we define the set*

$$\mathcal{M}_\delta^\varepsilon \stackrel{\text{def}}{=} \left\{\gamma(z) \ ; \ z \in \mathbb{R}^D, \|\Phi(\gamma(z)) - \Phi(x^\star)\|_2 \leq \varepsilon \text{ and } \|\gamma(z) - x^\star\|_2 \leq \delta\right\}. \tag{18}$$

and we refer to γ as the parameterization *of $\mathcal{M}_\delta^\varepsilon$.*

From Theorem 5, this set is well defined whenever Φ is of class $\mathcal{C}^1$. Moreover, the Jackpot set recovers the local structural non-identifiabilities (i.e., $\mathcal{U}_\delta^0$).

Theorem 8 *Let Φ be of class $\mathcal{C}^1$ and $D = \dim\ker J_\Phi(x^\star) > 0$. Moreover assume that $\dim\ker J_\Phi(x)$ remains constant in a neighborhood of $x^\star$. Then there exists $\delta > 0$, such that Jackpot recovers the uncertainty region locally, that is $\mathcal{M}_\delta^0 = \mathcal{U}_\delta^0$.*

All the preceding results hold under the assumption that Φ is of class $\mathcal{C}^1$. In this case, the set $\mathcal{M}_\delta^\varepsilon$ is well-defined, but establishing whether it admits a manifold structure is nontrivial and remains an open problem. However, when Φ is of class $\mathcal{C}^2$, one can additionally prove that $\mathcal{M}_\delta^\varepsilon$ indeed has a smooth manifold structure.

Theorem 9 *Let Φ be of class $\mathcal{C}^2$ and $D \geq \dim\ker J_\Phi(x^\star)$. Then $\mathcal{M}_\delta^\varepsilon$ defines a manifold.*

The proofs of Theorem 8 and 9 are provided in Appendix A.3.2.

2.2.2 Approximation Guarantees

In this section, we focus on the case Φ of class $\mathcal{C}^2$. We show in Theorem 12 that $\mathcal{M}_\delta^\varepsilon$ provides a good approximation of $\mathcal{U}_\delta^\varepsilon$.

Assumption 10 (A simple approximation condition) *Let*

$$\mathcal{S}_z = \left\{x \in \mathcal{U}_\delta^\varepsilon, V_D^{\star,T}(x - x^\star) = z\right\}$$

denote the slice of $\mathcal{U}_\delta^\varepsilon$ which has coordinate z on $\mathcal{T}_{x^\star}$. We assume that there exists $\eta > 0$ such that

$$\sup_{\substack{z \in \mathbb{R}^D \\ \|\gamma(z) - x^\star\|_2 \leq \delta}} \operatorname{diam}(\mathcal{S}_z) \leq \eta. \tag{19}$$

This assumption is illustrated in Fig. 4. Notice that the value $\eta > 0$ can be large even if the set is thin, when the set is bent a lot with respect to $\mathcal{T}_{x^\star}$. For a $\mathcal{C}^2$ mapping Φ, the thickness η can be controlled by regularity properties as shown in the following proposition, the proof of which is deferred to Appendix A.3.2.

Proposition 11 (Uniformly bounded curvature) *Assume that Φ is of class $\mathcal{C}^1$ and that $D \geq \dim\ker(J_\Phi(x^\star)) = N - R^\star$. Assume moreover that its Jacobian J_Φ is L-Lipschitz over the ball $\mathcal{B}(x^\star, \delta)$, i.e.*

$$\forall x, x' \in \mathcal{B}(x^\star, \delta), \quad \|J_\Phi(x) - J_\Phi(x')\|_{op} \leq L \left\|x - x'\right\|_2 \tag{20}$$

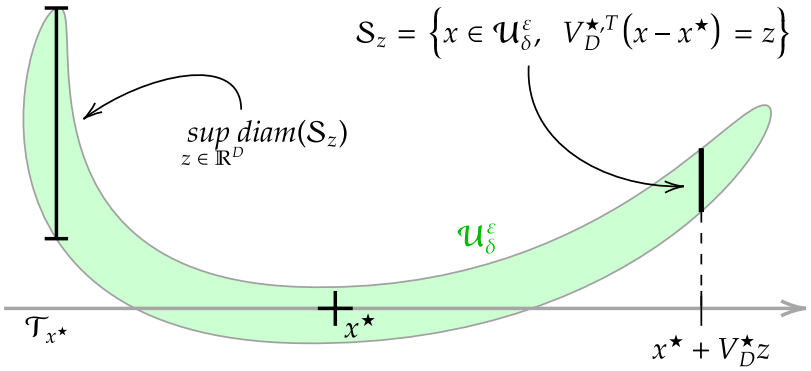


Figure 4: An illustration of Assumption 10. The slices $\mathcal{S}_z$ of $\mathcal{U}_\delta^\varepsilon$ have a diameter uniformly bounded by η.

where $\|\cdot\|_{op}$ *is the operator norm. Finally, suppose that the Lipschitz constant* L *satisfies*

$$L < \frac{\sigma^\star_{N-D}}{\delta}$$

where $\sigma^\star_{N-D}$ *is the* $(N-D)$*-th singular value of* $J_\Phi(x^\star)$*. Then Assumption 10 holds with*

$$\eta = \frac{2\varepsilon}{\sigma^\star_{N-D} - L\delta}. \tag{21}$$

From Proposition 11, we get that for a twice differentiable mapping Φ, Assumption 10 is always satisfied for sufficiently small δ (i.e., sufficiently locally). We are now in a position to show how well $\mathcal{M}_\delta^\varepsilon$ approximates $\mathcal{U}_\delta^\varepsilon$ in terms of the Hausdorff distance defined in (2). The proof is provided in Appendix A.3.2.

Theorem 12 (Approximation guarantees) *Let* Φ *be of class* $\mathcal{C}^2$ *and* $D \geq \dim\ker J_\Phi(x^\star)$*. Then, under Assumption 10,* $\mathcal{M}_\delta^\varepsilon$ *is a good approximation manifold of* $\mathcal{U}_\delta^\varepsilon$ *in the sense that:*

$$d_{\mathcal{H}}(\mathcal{M}_\delta^\varepsilon,\ \mathcal{U}_\delta^\varepsilon) \leq \eta, \tag{22}$$

where $d_{\mathcal{H}}$ *is defined in* (2)*. More precisely, we have*

$$\mathcal{M}_\delta^\varepsilon \subseteq \mathcal{U}_\delta^\varepsilon \subseteq \mathcal{M}_\delta^\varepsilon + \mathcal{B}(0,\eta). \tag{23}$$

Remark 13 *A few remarks are in order:*

- *In the linear case, since the Hessian is null, any radius* $\delta > 0$ *suits as long as the dimension* D *satisfies* $D \geq \dim\ker J_\Phi(x^\star)$*.*
- *An upper-bound on the Hessian norm* $\|H_\Phi\|_{2,\delta}$ *is*

$$\sup_{x \in \mathcal{B}(x^\star,\delta)} \left\| \left(\sigma_{max}\left(H_{\Phi_m}(x)\right)\right)_{m \leq M} \right\|_2$$

with Φ_m *being the* m*-th component of* Φ*.*

2.2.3 Numerical Computation

From a numerical perspective, we have to discretize the manifold $\mathcal{M}_\delta^\varepsilon$. This can be done by evaluating $(\mathcal{P}_z)$ on a discrete subset $Z = \{z_k\}_{0\leq k\leq K}$ of $\mathbb{R}^D$. We let $\mathcal{M}_\delta^\varepsilon(Z)$ denote the discretized manifold:

$$\mathcal{M}_\delta^\varepsilon(Z) \stackrel{\text{def}}{=} \mathcal{M}_\delta^\varepsilon \cap \{\gamma(z); z \in Z\}\,. \tag{24}$$

The main problem now reduces to evaluating γ for grid points Z satisfying the constraints. The principle is to propagate a front starting from $z = 0_D$ with a grid-search algorithm and to solve $(\mathcal{P}_z)$ at each z. A pseudo-code is given in Algorithm 1. We detail the two steps below.

Grid search In this work, we simply discretize $\mathbb{R}^D$ with a Cartesian grid. Other choices such as the honeycomb pattern are possible, but have not been explored. Fixing a grid length $s > 0$, we define the following Cartesian grid

$$Z_s \stackrel{\text{def}}{=} \left(s\mathbb{Z}^D\right) \cap \mathcal{B}_\infty(0_D, \delta) = s\left\{-\left\lfloor\frac{\delta}{s}\right\rfloor, \ldots, \left\lfloor\frac{\delta}{s}\right\rfloor\right\}^D. \tag{25}$$

Since the initial point $\gamma(0) = x^\star$ is known, the grid search should start with the point $0_D \in Z$. Solving $(\mathcal{P}_z)$ can be achieved with first order methods, which require a starting point. Since γ is a smooth mapping, a good approximation of $\gamma(z)$ is $\gamma(z')$ for some z' where γ has already been computed. This leads to consider a grid search method starting from $0_D \in Z$ and passing from neighbor to neighbor. The Breadth-First Search (BFS) algorithm is well suited for this task. We add a graph structure $G = (Z, E)$ (where E is the set of edges) on the grid in order to apply this algorithm. Given this structure, the natural choice of initial guess for the optimization at a non already-explored node is thus given by the closest already computed neighbor. BFS method also makes it possible to stop the search in the neighborhood of a vertex whenever one of the two criteria defining (24) are violated.

Optimization algorithm Each step of Algorithm 1 requires solving the optimization problem $(\mathcal{P}_z)$. Obtaining the following result is based on inserting the linear constraint in the objective function. Its proof is provided in Appendix A.4.1.

Proposition 14 *Let $\Pi_\perp \stackrel{\text{def}}{=} \mathrm{Id}_N - V_D^\star V_D^{\star,T}$ be the orthogonal projection on the subspace $(\mathrm{Im} V_D^\star)^\perp$ and consider the following optimization problem for $z \in \mathbb{R}^D$*

$$\eta_\perp(z) \stackrel{\text{def}}{=} \underset{x\in\mathbb{R}^N}{\text{argmin-loc}}\ \frac{1}{2}\left\|\Phi(x^\star + V_D^\star z + \Pi_\perp x) - \Phi(x^\star)\right\|_2^2. \tag{$\mathcal{P}'_z$}$$

Then there exists a local neighbor $\mathcal{V} \subseteq \mathbb{R}^D$ of 0_D such that the solution γ of $(\mathcal{P}_z)$ verifies for $z \in \mathcal{V}$,

$$\gamma(z) = x^\star + V_D^\star z + \Pi_\perp \eta_\perp(z). \tag{26}$$

Moreover, if Φ is of class $\mathcal{C}^2$, a gradient descent algorithm applied to $(\mathcal{P}'_z)$ converges linearly provided the step size is properly chosen (Polyak, 1987).

Algorithm 1 BFS parameterization of $\mathcal{M}_\delta^\varepsilon(Z)$

Inputs of the model:
- $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ a $\mathcal{C}^1$ mapping.
- $x^\star \in \mathbb{R}^N$ an estimation of input parameter.
- $1 \leq D < N$ the manifold dimension.
- $V_D^\star \in \mathbb{R}^{N\times D}$ the D lowest right singular vectors of $J_\Phi(x^\star)$.

Inputs of the paramaterization:
- $\delta > 0$ a radius around $x^\star$ (to deal with $\|x - x^\star\|_2 \leq \delta$).
- $\varepsilon > 0$ a discrepancy threshold (to deal with $\|\Phi(x) - \Phi(x^\star)\|_2 \leq \varepsilon$).
- $G = (Z, E)$ a connected graph with $0_D \in Z \subseteq \mathbb{R}^D$.

Initialization:
$\eta_\perp(0_D) \leftarrow 0_N$.
$\gamma(0_D) \leftarrow x^\star$.
repeat
  $(z, z_{\text{prev}}) \leftarrow$ the couple (non-evaluated, evaluated) neighboring vertices with smallest distance.
  $\eta_\perp(z) \leftarrow$ solution of $(\mathcal{P}'_z)$ initialized at $\eta_\perp(z_{\text{prev}})$ with an L-BFGS method.
  $\gamma(z) \leftarrow x^\star + V_D^\star z + (\mathrm{Id}_N - V_D^\star V_D^{\star,T})\eta_\perp(z)$.
  if $\|\Phi(\gamma(z)) - \Phi(x^\star)\|_2 > \varepsilon$ **or** $\|\gamma(z) - x^\star\|_2 > \delta$ **then**
    Remove vertex z from Z and all its edges from E.
  end if
until No couple (z, z_{prev}) exists
Return $\{\gamma(z)\}_{z\in Z}$

To further accelerate the computation of $\gamma(z)$ we propose to use the L-BFGS method on $(\mathcal{P}'_z)$, which is a quasi-Newton algorithm. It accelerates the convergence by using a preconditionning based on an estimation of the Hessian using the gradients at each step. It produces a sequence of iterates $(\gamma_k(z))$. A natural stopping criteria for this sequence is $\|\Phi(\gamma_k(z)) - \Phi(x^\star)\|_2 \leq \nu \cdot \varepsilon$ with $\nu < 1$ a small constant.

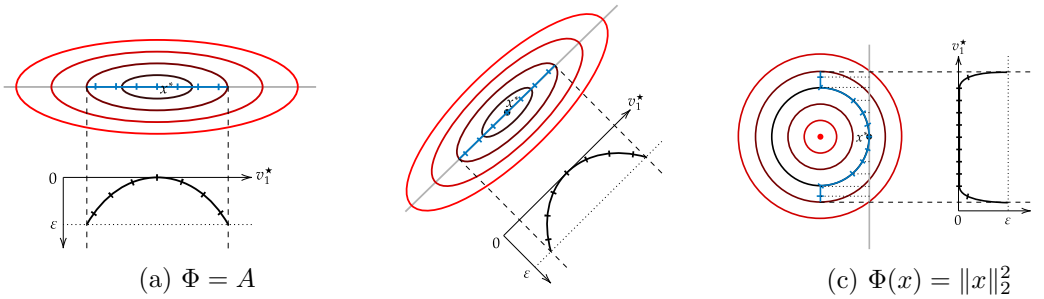
Moreover, we also deploy a warm-start strategy over the grid Z to further accelerate the convergence of L-BFGS, leveraging the smoothness of the manifold to obtain good initial points. We summarize the whole process and give a pseudo-code in Algorithm 1.

Approximation result One can extend the approximation result of the previous section to this discrete framework. The proof of the following corollary is provided in Appendix A.4.2.

Corollary 15 *Under the same assumptions as in Theorem 12, and using a Cartesian grid $Z = Z_s$ defined in* (25)*, the discretized manifold provides a good approximation of the uncertainty region. Specifically, we have*

$$d_{\mathcal{H}}(\mathcal{M}_\delta^\varepsilon(Z), \mathcal{U}_\delta^\varepsilon) \leq \eta + L\sqrt{D}\frac{s}{2}, \tag{27}$$

where $L > 0$ is the Lipschitz constant of the local solution γ to $(\mathcal{P}_z)$, $d_{\mathcal{H}}$ is the Hausdorff distance as defined in Definition 2, and $\eta > 0$ is the threshold defined in Theorem 12.



(a) $\Phi = A$ (b) $\Phi = UAU^T$ (c) $\Phi(x) = \|x\|_2^2$

Figure 5: Illustration of the proposed Jackpot method with $D = 1$ on the same examples as in Fig. 1. The low dimensional manifold $\mathcal{M}_\delta^\varepsilon$ is represented with blue lines. In contrast to the profile likelihood method, the profile is computed along the most suitable direction. In addition, the set $\mathcal{M}_\delta^\varepsilon$ is the best possible one-dimensional approximation of the 2 dimensional ellipses in (a) and (b). In example (c), the approximating set $\mathcal{M}_\delta^\varepsilon$ precisely coincides with the unit circle (i.e., $\mathcal{U}_\delta^\varepsilon$), except on the extremities.

2.2.4 Discussion

Before turning to the numerical experiments, we describe the strengths and weaknesses of the approach in order to highlight the problems for which it is best suited.

Behavior on the toy examples To begin with, let us illustrate how the algorithm behaves on the three introductory examples of Fig. 1. We see that the algorithm outputs one dimensional sets (in blue), which correctly approximate the uncertainty regions in all three cases. The method suffers neither from the limitation of the choice of coordinates as were profile likelihood methods, nor from the instability of Monte Carlo-based methods when the Jacobian of Φ is degenerate.

Adapting to large dimensions Since the LOBPCG algorithm is matrix-free and combined with an automatic differentiation algorithm, our algorithm is scalable to large model (up to millions of parameters). The approximation being of smallest dimension D, it can be represented quite easily at least for dimensions $D = 1, 2, 3$. Depending on the geometry of the problem, our method gives a full exploration of the uncertainty region or only a partial one as shown in Fig. 6.

When is it most useful? From our theoretical analysis in Section 2.2.2, one can see that the low-dimensional manifold $\mathcal{M}_\delta^\varepsilon$ is a good approximation of $\mathcal{U}_\delta^\varepsilon$ provided that D is sufficient large relatively to the number of "low" singular values of $J_\Phi(x^\star)$. Yet, for large-scale problems, it is not possible to set D too large due to computational limitations. This leads us to classify problems in three categories depending on the Jacobian singular spectrum, as illustrated in Fig. 6.

- When the Jacobian is well-conditioned ("flat" spectrum, Fig. 6.a), we have practical local identifiability and the Jackpot method will detect it.
- When the Jacobian is badly-conditioned, we distinguish two cases.

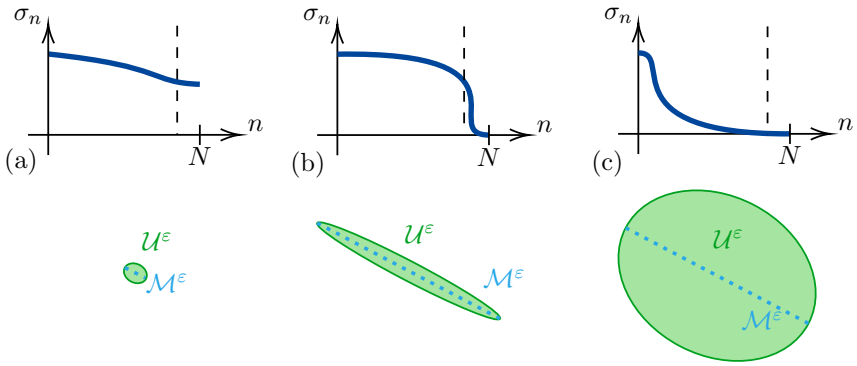


Figure 6: Relation between Jacobian singular spectrum and manifold approximation of the uncertainty region $\mathcal{U}^\varepsilon$. (a) No small singular values: the problem is identifiable. (b) Small number of small singular values. (c) Large number of small singular values. The set $\mathcal{U}^\varepsilon$ can not be approximated by a low dimensional manifold.

- If the spectrum admits few low singular values with a sharp transition (Fig. 6.b), $\mathcal{U}^\varepsilon_\delta$ can be well approximated by a low-dimensional manifold and the Jackpot method will compute it.
- Otherwise (e.g., Fig. 6.c), $\mathcal{U}^\varepsilon_\delta$ cannot be well approximated by a low-dimensional manifold. In this situation, the manifold $\mathcal{M}^\varepsilon_\delta$ computed by Jackpot (with "small" D) will only partially represent $\mathcal{U}^\varepsilon_\delta$.

Notice that in the last case, we cannot expect to "sketch" the uncertainty set efficiently due to its intrinsically high dimensional nature.

Stability The Jackpot algorithm depends heavily on the subspace spanned by the D lowest right singular vectors of the Jacobian $J_\Phi(x^\star)$. This subspace defines the tangent space of the constructed manifold at $x^\star$. The stability of this subspace to perturbations in $J_\Phi(x^\star)$ depends on the spectral gap between the D-th and $(D+1)$-th smallest singular values, that is, on the quantity $\sigma_{N-D+1} - \sigma_{N-D}$ (Stewart and Sun, 1990). When this gap is large, the D-dimensional subspace is well-separated from the remaining directions, and small errors in $J_\Phi(x^\star)$ (due, e.g., to numerical differentiation or stochastic estimation) lead to only small perturbations of the tangent space. Conversely, when the singular values around the D-th index are clustered or close to zero, the subspace becomes highly sensitive to errors in $J_\Phi(x^\star)$. In this situation, which is not favorable, as explained in the previous paragraph, many manifolds are near equivalent in terms of approximation, and Jackpot will just select one of them.

Adversarial versus natural perturbations Our goal in this paper is to design a manifold that approximates $\mathcal{U}^\varepsilon = \Phi^{-1}\left(\mathcal{B}\left(\Phi(x^\star), \varepsilon\right)\right)$ in Hausdorff distance. We call the resulting set an *adversarial manifold*. The directions $(v^\star_{N-D+1}, \dots, v^\star_N)$ represent the perturbation directions that most significantly affect the inverse map Φ^{-1}. These directions also capture the local geometry of $\mathcal{U}^\varepsilon$ most effectively, as seen in Theorem 12.

In contrast, Monte Carlo sampling methods serve a different purpose: they simulate "natural" perturbations, which are typically observed in real-world scenarios. Given a random

vector $Y = \Phi(x) + B$, where B follows some probability distribution μ, they capture the pushforward distribution $\Phi_\sharp^{-1}(\mu(\cdot - \Phi(x)))$. Points derived from the Jackpot method, however, may possess very low likelihood and would almost never appear through Monte Carlo sampling. This distinction becomes clearer in our numerical experiments.

Adversarial and natural perturbations offer distinct advantages, depending on the nature of uncertainty one aims to describe. Adversarial perturbations (Antun et al., 2020) are critical in scenarios where incorrect decisions may lead to severe consequences, as they identify the worst-case distortions. On the other hand, natural perturbations provide an averaged view of uncertainty, reflecting more common or benign variations.

The limitations of locality Our approximation and our analysis is only local. For instance, in Fig. 3, only half of the circle can be parameterized by our method. Hence the method should only be used locally. Solutions such as a periodic reassessment of the tangent plane could be considered, but we did not explore this rather computationally heavy approach in this paper.

Link with sensitivity analysis The uncertainty region is an object closely related but different from the *sensitivity region*, which can also be treated by a slightly modified Jackpot algorithm. Let $S : \mathbb{R}^M \to \mathbb{R}^N$ be a reconstruction mapping from the measurement space $\mathbb{R}^M$ to signal space $\mathbb{R}^N$. We define the sensitivity region as

$$\mathcal{S}^\varepsilon \stackrel{\text{def}}{=} S\left(\mathcal{B}(\Phi(x^\star), \varepsilon)\right), \tag{28}$$

which corresponds to the set of reconstructions obtained when perturbing the data around $\Phi(x^\star)$ with a noise of norm smaller than ε.

In the idealized case where S is a "perfect inverse", satisfying $\Phi(S(y)) = y$ for all $y \in \mathbb{R}^M$, it holds that

$$\mathcal{S}^\varepsilon \subseteq \mathcal{U}^\varepsilon. \tag{29}$$

That is, the sensitivity region is a subset of the full uncertainty region, reflecting additional constraints imposed by the solver, such as regularization or prior information. It captures the *residual non-identifiability* under the reconstruction method.

Both $\mathcal{U}^\varepsilon$ and $\mathcal{S}^\varepsilon$ provide complementary views of the estimation problem:

- The set $\mathcal{U}^\varepsilon$ quantifies the fundamental ambiguity in the inverse problem, independent of any reconstruction algorithm.
- The set $\mathcal{S}^\varepsilon$ reflects the actual spread of outputs produced by a specific solver when facing data perturbations.

The direct extension of the Jackpot algorithm to sensitivity analysis consists in defining the manifold:

$$\gamma(z) = \underset{y \in C(z)}{\text{argmax-loc}} \, \|S(y) - S(y^\star)\|_2^2, \tag{30}$$

where $C(z)$ is an affine subspace of dimension $M - D$, centered at $y^\star + V_D^\star z$, and oriented along the orthogonal complement of the subspace spanned by the D dominant right singular vectors of the Jacobian $J_S(y^\star)$. More precisely,

$$C(z) = y^\star + V_D^\star z + \text{ran}(V_D^\star)^\perp, \tag{31}$$

where $V_D^\star = [v_1^\star, v_2^\star, \dots, v_D^\star] \in \mathbb{R}^{M\times D}$ is the matrix formed by the top D right singular vectors of $J_S(y^\star)$, and $y^\star = \Phi(x^\star)$.

However, the optimization problem (30) may not admit a solution in general, as it involves maximizing over an unbounded set. To overcome this issue, we introduce a compact alternative:

$$C(z) = \left\{ y \in \mathbb{R}^M : \|y - y^\star\|_2 = \|z\|_2,\, y \in y^\star + \mathrm{cone}(V_D^\star z) + \mathrm{ran}(V_D^\star)^\perp \right\}, \tag{32}$$

where $\mathrm{cone}(V_D^\star z) = \{t \cdot V_D^\star z : t \in \mathbb{R}^+\}$ is a half-line in direction $V_D^\star z$. The resulting set $C(z)$ is compact: it is a spherical cap (half-sphere) of radius $\|z\|_2$, centered at $y^\star$, and intersected with an $(M+1-D)$-dimensional subspace. We choose the cone rather than the full span to capture the possible asymmetry of the sensitivity region around $y^\star$.

For intuition, consider the case $D = 1$. Then the curve $\gamma(z)$ corresponds to the direction of maximal sensitivity, i.e., the worst-case adversarial perturbation at fixed radius. More precisely, we have:

$$\{\gamma(z), \gamma(-z)\} \ni \underset{\|y-y^\star\|_2=\|z\|_2}{\text{argmax-loc}} \|S(y) - S(y^\star)\|_2^2. \tag{33}$$

Up to this point, we have used the definition of $C(z)$ in (32). This alternative compact version is equally suitable for both uncertainty and sensitivity analysis. For clarity of exposition, we chose to present the former; however, most of the theoretical analysis and practical conclusions apply to the latter as well.

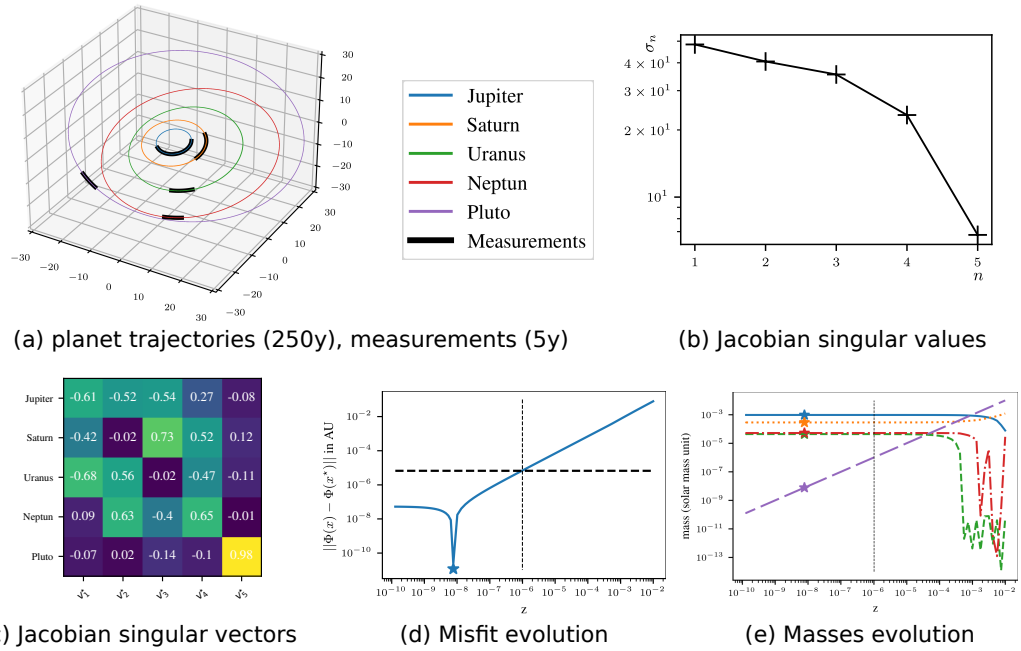
3. Numerical Experiments

In this section, we provide a few numerical examples on complex problems and compare it with some alternatives in the main three cases of practical identifiability described in Section 1.1.

3.1 Measuring Masses in the Solar System

In this first example, we address the problem of identifying the masses of five planets of the solar system (Jupiter, Saturn, Uranus, Neptun, and Pluto), from their approximate positions at a few time points. We assume that the planets are moving according to Newton's law of gravitation. This corresponds to the problem described in Section 1.1.1.

Notation We sort the planets with respect to their distance to the Sun. We let $1 \le n \le N$ denote the index of the n-th outer planet and associate the index 0 to the Sun. We let $u_n(t) \in \mathbb{R}^3$ denote the positions in AU of the n-th planet at time t, in an heliocentric frame. Therefore $u_0(t) = 0$ for all time. We let w_n denote the weight of the n-th planet in the solar mass unit, that is $w_0 = 1$. Moreover, we let $u(t) = (u_1(t), \dots, u_N(t)) \in \mathbb{R}^{N\times 3}$ denote the vector of positions and $w = (w_1, \dots, w_N)$ denote the vector of masses. The time is expressed in years. The time $t = 0$ corresponds to January 1, 2000. Initial positions and speeds are taken from IMCCE (2019).



(a) planet trajectories (250y), measurements (5y)

(b) Jacobian singular values

(c) Jacobian singular vectors

(d) Misfit evolution

(e) Masses evolution

Figure 7: Solar system experiment with Jackpot. After measuring the approximate planet's positions for 5 years, we wish to recover their masses. The spectrum of the Jacobian (b) indicates that a direction is particularly uncertain. Looking at the Jacobian singular vectors (c), we see that it corresponds to Pluto's mass. In (d) and (e), we see the evolution of the misfit and of the masses along the 1D set $\mathcal{M}_\delta^\varepsilon$, and observe that all masses are stable (relative to their mass), except Pluto's which can vary by 4 orders of magnitude.

Modeling The solar system can be modeled through the second order dynamical system:

$$\begin{cases} u_0(t) = 0 \\ \ddot{u}_n(t) \overset{\text{def}}{=} -\sum_{n'=0}^{N} \dfrac{G w_{n'}}{\|u_n(t) - u_{n'}(t)\|_2^3}(u_n(t) - u_{n'}(t)), \quad 1 \le n \le N. \end{cases} \tag{34}$$

To discretize this system, we use a Runge-Kutta scheme of order 4 (Butcher, 2016). This yields a forward mapping Φ of the form:

$$\Phi : w \mapsto (u(t_k))_{0 \le k \le K}, \tag{35}$$

where $t_k = k\Delta t$ denote uniformly spaced sampling points. The measurements are taken on a period of 5 years within intervals of $\Delta t = 7$ days, leading to $K = 260$ positions for each planet. Notice that 5 Earth-years do not cover a complete revolution for the outer planets. For instance, Pluto's revolution lasts about 250 years. This is illustrated on Fig. 7 (a), where the black lines illustrate the measurements. We add white Gaussian noise to the measurements with a standard deviation $\varepsilon = 1000$ km ($\simeq 7 \cdot 10^{-6}$ AU, where AU is the Earth-Sun distance). This yields a measurements vector $y = \Phi(\bar{w}) + b$, where $\bar{w}$ denotes the true planets' weights and b is the random perturbation.

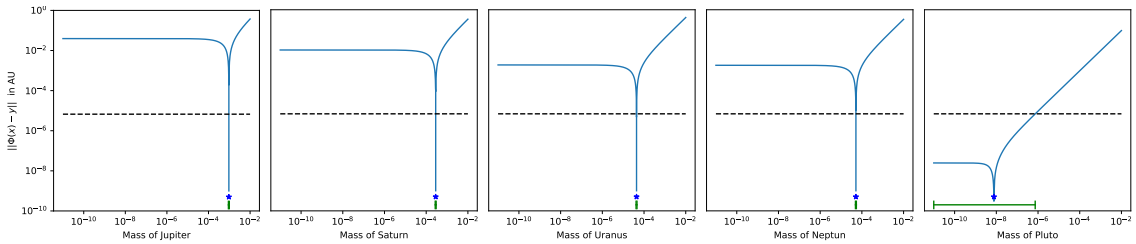


Figure 8: Profile likelihood on the solar system problem. The blue stars represent the true values of the parameters. Green segments represent confident interval. Black dashed lines represent the error threshold ε. Profile likelihood detects the Pluto's mass non-identifiability.

Uncertainty with Jackpot To estimate the masses w, we first solve the nonlinear problem

$$w^\star = \operatorname*{argmin}_{w\in\mathbb{R}^N} \frac{1}{2}\|\Phi(w) - y\|_2^2 \tag{36}$$

using a gradient descent to high accuracy. The gradient is evaluated through automatic differentiation.

This gives a fairly good estimate of Pluto's mass, overestimating the true estimate by 4%. Then, we run algorithm 1. The singular values of the Jacobian of Φ are displayed in Fig. 7 b). As can be seen, the last singular value is nearly ten times smaller than the others. Taking a close look at the singular vectors in Fig. 7 c), shows that the last singular vector is entirely concentrated on Pluto's mass, indicating that the estimation of this planet's mass is dubious.

We then compute a $D=1$-dimensional manifold approximation $\mathcal{M}_\delta^\varepsilon$ of the uncertainty region with the Jackpot method. As can be seen in Fig. 7, we can vary the range of Pluto's mass within the range $[10^{-10}, 10^{-6}]$, while staying consistent with the data. In that range, the other planets masses nearly do not change (at least relative to their weight). This illustrates a huge uncertainty on the mass of Pluto when evaluating it with 5 years of planetary observations only.

Comparisons with other approaches In Fig. 8 we use profile likelihood on this problem to estimate the uncertainty. This problem is particularly well suited to this approach, since only Pluto's mass is uncertain. In this case, we see that all profiles are concentrated on the true planets mass, apart for Pluto. There, the profile likelihood method recovers the same uncertainty interval as the one provided by Jackpot. The only difference is that Jackpot was able to automatically find this direction, without having to explore all of them. Also notice that profile-likelihood methods would likely bring partial information on the uncertainty if Pluto was discarded from the set. Indeed, the top-left 4x4 block in Fig. 7 c) indicates strong correlation effects between the different weights.

We do not show the results of the Monte Carlo sampling method since it fails for this example. Indeed, the sampling process only provides points that are very close to $x^\star$, and the method fails to recover the uncertainty set. This phenomenon is similar to the one observed in Fig. 2 (c): the regression problem to recover the samples x^i is very flat along the direction of Pluto's mass.

3.2 Blind Deblurring

In this section, we explore the problem described in Section 1.1.2. Observing a biological object with a microscope yields diffraction limited (i.e. blurry) images. When the impulse response, or Point Spread Function (PSF) of the system is unknown, improving the image sharpness requires solving a blind deblurring problem. In this section, we showcase how Jackpot makes it possible to evaluate uncertainty on the recovered blur kernels and images.

Modeling We consider the classical image formation model

$$y = k(\theta) * x + b, \tag{37}$$

where $x \in \mathbb{R}^M$ denotes the observed sample, $*$ denotes the convolution product, $b \in \mathbb{R}^M$ is a noise vector, and $k : \mathbb{R}^N \to \mathbb{R}^M$ represents the point spread function (PSF) of the system. It depends on parameters $\theta \in \mathbb{R}^N$. The scalar theory of diffraction (Goodman, 2005), informs us that the PSF $k(\theta)$ can be characterized by its pupil function; a two-dimensional function supported on a disk whose radius depends on the numerical aperture of the objective and the wavelength of the collected light. A common practice is to expand this pupil function on a truncated Zernike polynomial basis (Goodman, 2005). With this model, only a few Zernike polynomials are sufficient to describe a rich class of realistic PSF that includes typical aberrations. Hence, in our model (37), $\theta \in \mathbb{R}^N$ denotes the vector of Zernike coefficients. We fix $N = 8$ in this work, corresponding to the following optical aberrations: defocus, vertical and oblique astigmatism, trefoil and coma and primary spherical. The blind inverse problem consists in estimating both θ and x from y.

To that end, we follow the approach proposed by Gossard and Weiss (2024). We first train a reconstruction mapping $R(\theta, y)$ which computes an estimate $\hat{x}(\theta)$ such that the pair $(\theta, \hat{x}(\theta))$ is coherent with the data y. In this experiment, R is a deep plug-and-play image restoration method (DPIR). It corresponds to a half-quadratic splitting algorithm where the proximal step is replaced by a DRUNet denoiser with pre-trained weights (Zhang et al., 2021). Equipped with this (non-blind) reconstruction mapping R, we want to find the pair $(\theta, \hat{x}(\theta))$ that best fits the data y. That is, we want to minimize $\|k(\theta) * R(\theta, y) - y\|_2^2$. Using a Bayesian formalism, this means that we are trying to find the maximum a posteriori pair $(\hat{\theta}, \hat{x}(\hat{\theta}))$. This blind deblurring problem can be cast in our framework by using the following forward mapping

$$\Phi : \theta \mapsto k(\theta) * R(\theta, y). \tag{38}$$

In the first row of Fig. 9 we display the input image (a) as well as the noiseless measurements (b) we generated for this experiment.

Uncertainty with Jackpot Given y, we first estimate the parameters $\theta^\star$ by solving

$$\theta^\star = \underset{\theta \in \mathbb{R}^N}{\operatorname{argmin}} \frac{1}{2} \|\Phi(\theta) - y\|_2^2 \tag{39}$$

using a gradient descent with automatic differentiation. In Fig. 9 (c) and (d) we show the associated PSF $k(\theta^\star)$ and deconvolved image $\hat{x}(\theta^\star) = R(\theta^\star, y)$. Note that the image seems significantly better resolved than the observation, which suggests that reconstruction mapping R is constructed carefully.

Then, we run Algorithm 1. The singular values of the Jacobian of Φ and the two last singular vectors are displayed in Fig. 9 (e) and (f), respectively. As opposed to the previous experiment, we see no clear gap in the amplitude of the last singular values. The singular vectors are clearly mixing the different Zernike coefficients, suggesting that the profile likelihood methods will struggle indicating the main directions of uncertainty.

We then compute a $D=2$-dimensional manifold approximation $\mathcal{M}_\delta^\varepsilon$ of the uncertainty region with the Jackpot method. We obtain a parameterization of the Zernike coefficients vector $\theta(z)$ for $z \in \mathbb{R}^2$ with $\theta(0) = \theta^\star$. In Fig. 9 (g), we report the SNR between the outputs $\Phi(\theta(z))$ and $\Phi(\theta^\star)$. The level line of 40dBs is displayed in green.

In Fig. 11 we show how the profile likelihood behaves on this problem. The confidence intervals obtained with the profile likelihood method are also displayed as blue segments in Fig. 9 (g). They have been projected on the tangent plane $\mathcal{T}_{\theta^\star}$ for comparison. As can be seen from this plot, the profile likelihood method significantly underestimates the uncertainty domain, since the blue segments tips are far from the level line boundaries.

In Fig. 10, we provide some zoomed regions of the deblurred images in (a) and of the corresponding PSF in (b) for different parameters $\theta(z) \in \mathcal{M}_\delta^\varepsilon$. The corresponding parameters z are shown as red stars in Fig. 9 (g). We see that it is possible to significantly deform the PSFs with a negligible modification of the measurements. We can also observe on Fig.s 10 (a) and (b) that some structures of the sample can be hallucinated without affecting the measurements too much. Overall, this shows that the studied blind deblurring process is unstable, though it produces sharp and nice looking images for all parameters.

Finally, let us mention that the Monte Carlo sampling method strongly underestimates uncertainty regions as illustrated in Fig. 12. The reason is related to the fact that we wish to recover a relatively low dimensional manifold (less than 8), but add noise on a high dimensional space of color images ($200 \times 200 \times 3$).

3.3 Posterior Exploration for an Image Deblurring Problem

In contrast to the previous section, we tackle regular inverse problems where the forward operator A is known perfectly.

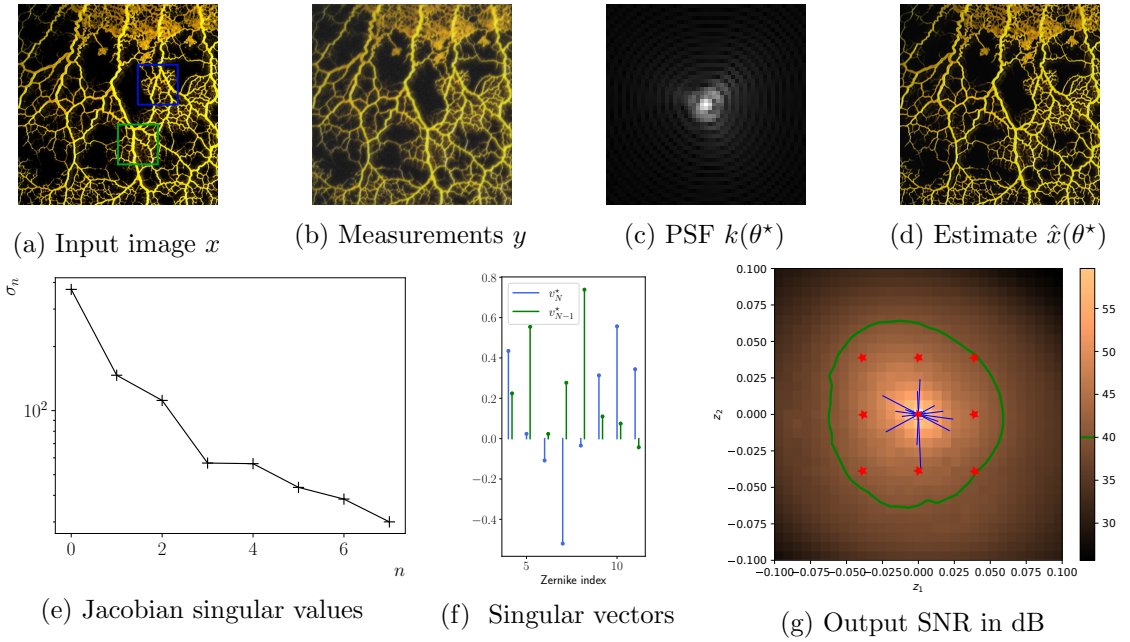
Formalism We explore the problem described in Section 1.1.3. Given $y = Ax + b$, with $b \sim \mathcal{N}(0, \sigma^2 \mathrm{Id})$, we construct the MAP estimator

$$\begin{aligned}\hat{x}_{\mathrm{MAP}} &\overset{\mathrm{def}}{=} \operatorname*{argmin}_{x \in \mathbb{R}^N} -\log p_{\mathbb{x}|\mathbb{y}}(x|y) \\ &= \operatorname*{argmin}_{x \in \mathbb{R}^N} \frac{1}{2\sigma^2} \|Ax - y\|_2^2 - \log p_{\mathbb{x}}(x).\end{aligned}$$

By first order optimality conditions, $\hat{x}_{\mathrm{MAP}}$ satisfies

$$\nabla \log p_{\mathbb{x}|\mathbb{y}}(\hat{x}_{\mathrm{MAP}}|y) = \frac{1}{\sigma^2} A^T(A\hat{x}_{\mathrm{MAP}} - y) - \nabla \log p_{\mathbb{x}}(\hat{x}_{\mathrm{MAP}}) = 0. \tag{40}$$

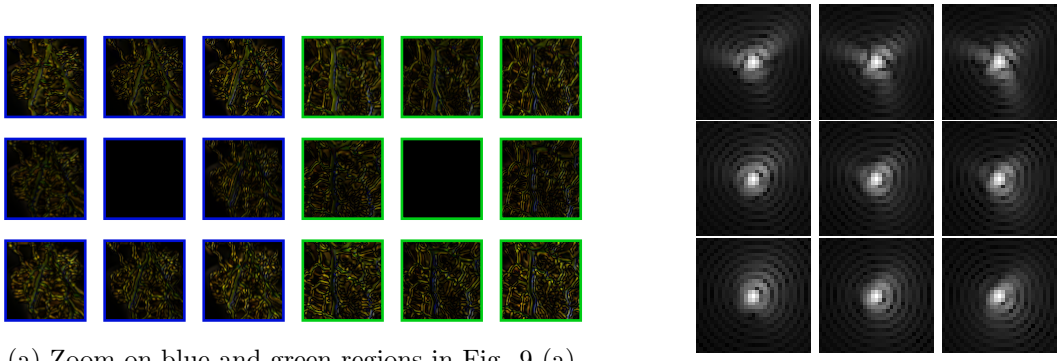
The term $\nabla \log p_{\mathbb{x}}(\hat{x}_{\mathrm{MAP}})$ can be estimated using a learned denoiser and Tweedie formula (Ho et al., 2020). In this paper, we use the denoiser proposed by Hurault et al. (2021), which enables to compute both the log likelihood and its gradient.



(a) Input image x (b) Measurements y (c) PSF $k(\theta^\star)$ (d) Estimate $\hat{x}(\theta^\star)$

(e) Jacobian singular values (f) Singular vectors (g) Output SNR in dB

Figure 9: Blind delurring experiment. The reconstructed image $\hat{x}(\theta^\star) = R(\theta^\star, y)$ in (d) seems significantly sharper than the observation y. There is no clear gap for the lowest singular values in (e), suggesting that the uncertainty domain has a rather large intrinsic dimension. In (f), the Zernike coefficients of the two last singular vectors of $J^\star$ are mixed, meaning that the main uncertainty directions are not aligned with pure optical abberations. In (g), we evaluated the function $\mathrm{SNR}(\Phi(\theta), \Phi(\theta^\star))$ for $\theta \in \mathcal{M}_\delta^\varepsilon$. The level line of 40dB, corresponding to a high fidelity is displayed as a green curve. We also projected the confidence intervals obtained with the profile likelihood method as blue segments.



(a) Zoom on blue and green regions in Fig. 9 (a)

(b) PSF $k(\theta(z))$

Figure 10: *Blind deblurring identifiability* - Grid sampling of the 2-dimensional manifold $\mathcal{M}_\delta^\varepsilon$. (a-b) Grid sampling of image differences $|\hat{x}(\theta(z)) - x^\star|$. (c) PSF grid-sampling $k(\theta(z))$.

Following Section 1.1.3, we set $\Phi = -\nabla \log p_{\mathbb{x}|\mathbb{y}}(\cdot|y)$. The uncertainty region $\mathcal{U}_\delta^\varepsilon$ now can be interpreted as the set of images with smallest gradient amplitude. We compute $\hat{x}_{\mathrm{MAP}}$ using a gradient descent run for many iterations to reach a near 0 gradient.

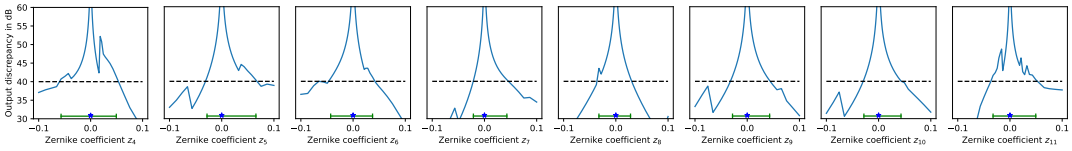
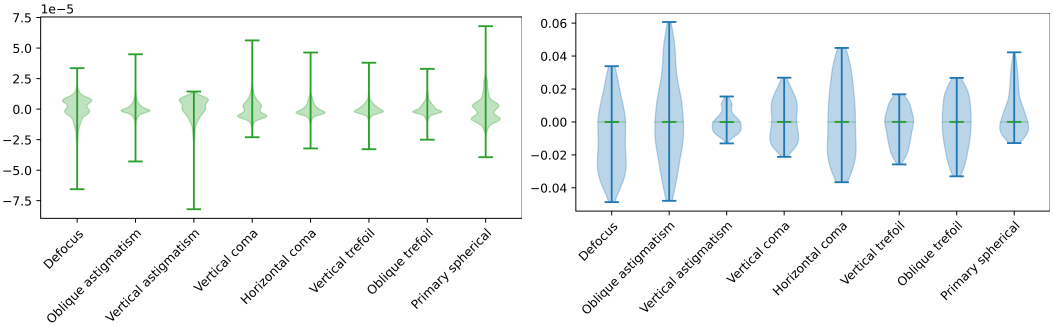


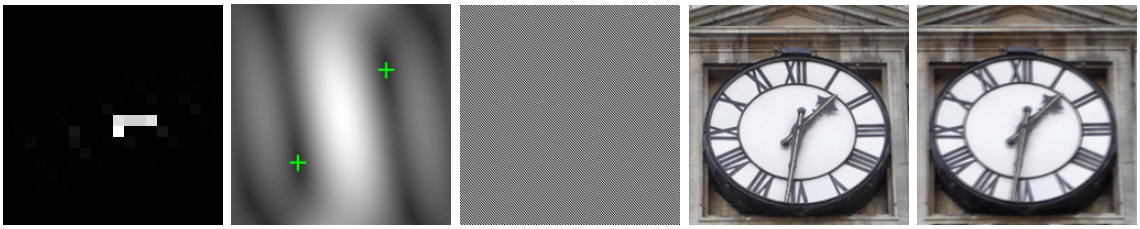
Figure 11: Profile likelihood on blind inverse problem. The blue stars represent the true values of the parameters. The green segments represent confidence intervals. The two first coefficients seem less identifiable than the other from this analysis.



(a) Monte Carlo sampling

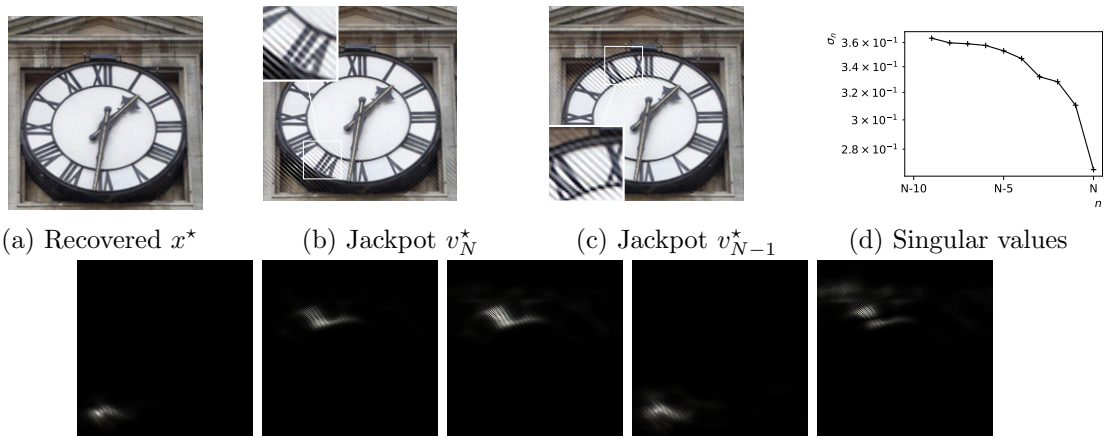
(b) Jackpot

Figure 12: Violinplots of the Zernike coefficients obtained using a Monte Carlo sampling with 1000 points (left) and the Jackpot algorithm (right) for the blind deblurring experiment. Observe that the amplitudes obtained with Monte Carlo sampling are about 10^3 times lower than those obtained with Jackpot, illustrating the large discrepancy between natural and adversarial perturbations.



(a) PSF (b) PSF spectrum (c) Kernel element (d) Reference (e) Blurry image y

Figure 13: The deblurring problem. To illustrate Jackpot, we designed a specific PSF with a highly oscillatory 2D kernel (null-space). An element is shown in (c).



(a) Recovered $x^\star$ (b) Jackpot $v_N^\star$ (c) Jackpot $v_{N-1}^\star$ (d) Singular values

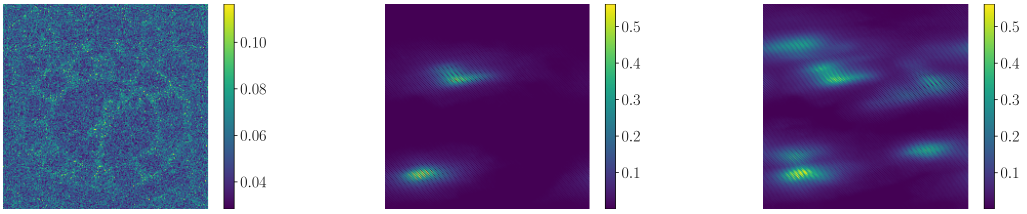
(e) The last 5 leftmost eigenvectors of $J^\star$ for the deblurring inverse problem.

Figure 14: Recovered image $x^\star$, 2 images recovered by Jackpot, the singular spectrum and 5 leftmost eigenpairs of $J^\star$.

Jackpot results We treat the 200×200 color image shown on Fig. 13d. The forward operator A is defined as a convolution with the PSF shown on Fig. 13a. Its discrete Fourier transform is shown on Fig. 13b. It possesses only two zeros (green crosses). Considering the Fourier transform symmetries for real signals, it means that the kernel is 2D for each color channel and therefore $\dim(\ker(A)) = 6$. This deblurring problem can be considered as mildly ill-posed.

The resulting blurry image and recovered images are shown on Fig. 13e and 14a respectively. The five last singular values of the Jacobian matrix $J^\star$ are displayed in Fig. 14d. Surprisingly, they all possess roughly the same amplitude. We could expect that the only directions of uncertainty are unlocalized patterns such as the one in Fig. 13c. Looking closer at the associated singular vectors in Fig. 14e, we see that they all consist of localized and highly oscillatory patterns. The reason why the uncertainty patterns can be localized is that we do not constrain $Ax = y$, but just promote it through the data term $\frac{1}{2\sigma^2}\|Ax - y\|_2^2$, leaving space for more uncertainty.

In this examples, profile-likelihood methods are unusable due to the too large number of parameters ($3 \times 200 \times 200$). We propose a comparison with Monte Carlo sampling in Fig. 15.



(a) Monte Carlo sampling (b) Jackpot with $D = 2$ (c) Jackpot with $D = 10$

Figure 15: Comparison between Monte Carlo sampling and Jackpot for image deblurring. The pixel-wise maximum distance with $x^\star$ is displayed for 1000 Monte Carlo points in (a) and for 20 Jackpot points and $D = 2$ in (b) and 100 Jackpot points and $D = 10$ in (c). Notice the different amplitude ranges.

After drawing 1000 samples, we display maximal pixel-wise distance to $x^\star$ both for the Monte Carlo and the Jackpot methods. As can be seen, the Monte Carlo method significantly underestimates the size of the uncertainty region. This again illustrates the difference between the effect of natural (Monte Carlo) and adversarial (Jackpot) perturbations.

Acknowledgments

This work was supported by the ANR Micro-Blind (grant ANR-21-CE48-0008), the University Research School EUR-MINT (ANR-18-EURE-0023). This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011012210R2). Photo credit: David Villa ScienceImage, CBI, CNRS.

Appendix A. Proofs

A.1 Definition and Characterization of Manifolds

Definition 16 *Let x be a point of an open set $\mathcal{U}$ of $\mathbb{R}^D$. A map $f : \mathcal{U} \to \mathbb{R}^N$ is said to be an* immersion at x *if f is of class $\mathcal{C}^1$, and its Jacobian $J_f(x)$ is injective. If this is true for all $x \in \mathcal{U}$, the map f is said to be an* immersion.

Definition 17 (Local definition by image of immersion, Lee, 2012) *Let $\mathcal{M}$ be a subset of $\mathbb{R}^N$. The set $\mathcal{M}$ is said to be a* manifold *of dimension D if, for all $x \in \mathcal{M}$, there exist*

- *a neighborhood $\mathcal{U}$ of 0_D in $\mathbb{R}^D$*
- *a neighborhood $\mathcal{V}$ of x in $\mathbb{R}^N$*
- *$f : \mathcal{U} \to \mathbb{R}^N$, an immersion at 0_D such that $f(0_D) = x$*

such that $f : \mathcal{U} \to \mathcal{V} \cap \mathcal{M}$ is an homeomorphism.

The proposed Jackpot builds an approximation $\mathcal{M}_\delta^\varepsilon$ (see Definition 7) of $\mathcal{U}_\delta^\varepsilon$ through the parametrization $\gamma : \mathbb{R}^D \to \mathbb{R}^N$ in $(\mathcal{P}_z)$. According to Definition 17, to prove that $\mathcal{M}_\delta^\varepsilon$ presents a manifold structure, we require the following two ingredients :

- γ is of class C^1 and $J_\gamma(0_D)$ is injective (i.e., γ is an immersion at 0_D), such that $\gamma(0_D) = x^\star$ (true by definition).
- there exist neighborhoods $\mathcal{V}$ of $x^\star$ in $\mathbb{R}^N$ and $\mathcal{U}$ of 0_D in $\mathbb{R}^D$ such that $\gamma : \mathcal{U} \to \mathcal{V} \cap \mathcal{M}$ is an homeomorphism.

Actually, the second point is implied by the first one as stated by Proposition 18 below. Consequently, the proof of the manifold structure of $\mathcal{M}_\delta^\varepsilon$ in Theorem 12 reduces to proving only the first point above.

Proposition 18 (Injective Jacobian implies a local structure of manifold) *Let $f : \mathbb{R}^D \to \mathbb{R}^N$ be a function of class $\mathcal{C}^1$. Assume that $J_f(0_D)$ is injective. Then there exists a neighborhood $\mathcal{U} \subset \mathbb{R}^D$ of 0_D such that $f(\mathcal{U})$ is a manifold.*

Proof First of all, J_f is locally injective near 0_D because of the lower semi-continuity of the rank function. Indeed, the set of points on which J_f is injective can be reformulated as the inverse image of the open set $\mathbb{R}^*$ under the continuous map $\det(J_f^T J_f)$ and thus is open.

From rank theorem (Lee, 2012, Thm. 5.13), up to diffeomorphisms, f is locally equal to its Jacobian which is locally injective, so is f. We actually get that f is locally an immersion.

From (Lee, 2012, Lem. 5.34), since f is locally an immersion, f is also locally an embedding on a neighborhood $\mathcal{U}$.

Finally from (Lee, 2012, Thm. 5.31), $f(\mathcal{U})$ is an embedded submanifold as the image of a smooth embedding, which concludes the proof. ■

A.2 Linear Approximation

A.2.1 Proof of Proposition 4

Defining the ellipsoid $\mathcal{E} \stackrel{\text{def}}{=} \{x \in \mathbb{R}^N \ ; \ \left\|\Sigma^\star V^{\star,T}(x - x^\star)\right\|_2^2 \leq \varepsilon^2\}$ we can reformulate

$$\widetilde{\mathcal{U}}_\delta^\varepsilon = \mathcal{E} \cap \mathcal{B}(x^\star, \delta) \text{ and } \widetilde{\mathcal{M}}_\delta^\varepsilon = \mathcal{E} \cap \mathcal{B}(x^\star, \delta) \cap \mathcal{T}_{x^\star}. \tag{41}$$

As $\widetilde{\mathcal{M}}_\delta^\varepsilon = \widetilde{\mathcal{U}}_\delta^\varepsilon \cap \mathcal{T}_{x^\star} \subseteq \widetilde{\mathcal{U}}_\delta^\varepsilon$, the Hausdorff distance is given by $d_{\mathcal{H}}(\widetilde{\mathcal{M}}_\delta^\varepsilon, \widetilde{\mathcal{U}}_\delta^\varepsilon) = \sup_{x \in \widetilde{\mathcal{U}}_\delta^\varepsilon} d(x, \widetilde{\mathcal{M}}_\delta^\varepsilon)$. Without considering the constraint $x \in \mathcal{B}(x^\star, \delta)$, we get the same for $\mathcal{E}$ and $\mathcal{E} \cap \mathcal{T}_{x^\star}$

$$d_{\mathcal{H}}(\mathcal{E} \cap \mathcal{T}_{x^\star}, \mathcal{E}) = \sup_{x \in \mathcal{E}} d(x, \mathcal{E} \cap \mathcal{T}_{x^\star}). \tag{42}$$

Now, we can consider the following equivalent optimization problem

$$\sup_{x \in \mathcal{E}} \frac{1}{2} \left\|(\mathrm{Id}_N - \Pi_{\mathcal{V}_D^\star})(x - x^\star)\right\|_2^2 \tag{43}$$

as $d(x, \mathcal{E} \cap \mathcal{T}_{x^\star})^2 = \left\|(\mathrm{Id}_N - \Pi_{\mathcal{V}_D^\star})(x - x^\star)\right\|_2^2$ where $\mathcal{V}_D^\star$ is defined in Equation 13. The associated Lagrangian is given by

$$\mathcal{L}(\lambda, x) = \frac{1}{2} \left\|(\mathrm{Id}_N - \Pi_{\mathcal{V}_D^\star})(x - x^\star)\right\|_2^2 - \frac{\lambda}{2} \left(\left\|\Sigma^\star V^{\star,T}(x - x^\star)\right\|_2^2 - \varepsilon^2 \right). \tag{44}$$

And its gradient is

$$\nabla \mathcal{L}(\lambda, x) = \begin{pmatrix} \sum_{n=1}^{N} s_n^2 (\sigma_n^\star)^2 - \varepsilon^2 \\ \sum_{n=1}^{N-D} s_n v_n^\star - \lambda \sum_{n=1}^{N} (\sigma_n^\star)^2 s_n v_n^\star \end{pmatrix} \tag{45}$$

where we denote the coordinates $x - x^\star = \sum_{n=1}^{N} s_n v_n^\star$. The optimum is achieved where the gradient vanishes, that is

$$\sum_{n=1}^{N} s_n^2 (\sigma_n^\star)^2 = \varepsilon^2 \ ; \ \begin{cases} s_n = 0, & \forall n \geq N - D + 1 \\ s_n(1 + \lambda \sigma_n^\star) = 0, & \forall n \leq N - D \end{cases}.$$

From this, only one s_n is non zero. Among the $N - D$ local optima, only the following is global as long as $\sigma_{N-D}^\star < \sigma_{N-D-1}^\star$ (otherwise, it is still a global optimum but not unique)

$$s_n = 0, \ \forall n \neq N - D \ ; \ s_{N-D} = \frac{\varepsilon}{\sigma_{N-D}^\star} \text{ and } \lambda = \frac{1}{\sigma_{N-D}^\star}. \tag{46}$$

Finally let's add the constraint $\|x - x^\star\|_2 \leq \delta$. If $\delta > \frac{\varepsilon}{\sigma_{N-D}^\star}$, then the supremum is achieved at $x^\star + \frac{\varepsilon}{\sigma_N^\star} v_{N-D}^\star$. Otherwise, $\delta < \frac{\varepsilon}{\sigma_{N-D}^\star}$ and the supremum is achieved at $x^\star + \delta v_{N-D}^\star$. In both cases, the Hausdorff distance verifies Equation 15.

A.3 Nonlinear Approximation

All proof are provided in a more general setting with the parametrization

$$\gamma(z) \stackrel{\text{def}}{=} \underset{A(x-x^\star)=z}{\text{argmin-loc}} \ \frac{1}{2} \|\Phi(x) - \Phi(x^\star)\|_2^2, \tag{$\mathcal{P}_A$}$$

where $A \in \mathbb{R}^{D \times N}$ with $1 \leq D \leq N$ satisfies the following assumption.

Assumption 19 *The matrix A verifies $AA^T = \mathrm{Id}_D$ and the matrix $\Pi_{\mathrm{im}A^T} + \Pi_{\ker A} \cdot J_\Phi(x^\star)^T J_\Phi(x^\star)$ is invertible. We denote the projections operators on the image and kernel of A as $\Pi_{\mathrm{im}A^T} \stackrel{\text{def}}{=} A^T A$ and $\Pi_{\ker A} \stackrel{\text{def}}{=} \mathrm{Id}_N - A^T A$ respectively.*

The results provided in the main text are specific instances of the results reported below with $A = V_D^{\star,T}$ for which Assumption 19 holds.

Lemma 20 *Assumption 19 holds true for $A = V_D^{\star,T}$.*

Proof With $A = V_D^{\star,T}$, the matrix

$$\Pi_{\mathrm{im}A^T} + \Pi_{\ker A} \cdot J_\Phi(x^\star)^T J_\Phi(x^\star) = V_D^\star \ \mathrm{Diag}\left(\sigma_1^2, \cdots, \sigma_{N-D}^2, 1, \cdots, 1\right) V_D^{\star,T}$$

is invertible by construction of $V_D^{\star,T}$. ■

A.3.1 Proof of Theorem 5

Theorem 5 is a specific instance of Theorem 21 below with $A = V_D^{\star,T}$.

Theorem 21 *Let $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ be a $\mathcal{C}^1$ map, $x^\star \in \mathbb{R}^N$, and $A \in \mathbb{R}^{D\times N}$ satisfying Assumption 19. Then, there exists an open neighborhood $\mathcal{V} \subseteq \mathbb{R}^D$ such that $(\mathcal{P}_A)$ admits a unique solution $\gamma : \mathcal{V} \to \mathbb{R}^N$ verifying $\gamma(0_D) = x^\star$. Moreover, it has the following properties*

- *If Φ is of class $\mathcal{C}^2$, then γ is of class $\mathcal{C}^1$. Moreover, its Jacobian is given by*

$$J_\gamma(z) = \left[\Pi_{\mathrm{im}A^T} + \Pi_{\ker A} \cdot \nabla^2 F(\gamma(z))\right]^{-1} A^T. \tag{47}$$

- *If Φ is of class $\mathcal{C}^1$ and J_Φ is locally Lipschitz and definable, then γ is a locally Lipschitz definable function. Moreover, we can express a conservative Jacobian for γ as*

$$\tilde{J}_\gamma : z \rightrightarrows \left\{ [\Pi_{\mathrm{im}A^T} + \Pi_{\ker A} \cdot B]^{-1} A^T \ ; \ B \in \tilde{J}_{\nabla F}(\gamma(z)) \right\}. \tag{48}$$

where $F(x) = \frac{1}{2}\left\|\Phi(x) - \Phi(x^\star)\right\|_2^2$ denotes the objective function of $(\mathcal{P}_A)$.

Proof Starting from the Karush-Kuhn-Tucker (KKT) conditions of Problem $(\mathcal{P}_A)$, the idea of the proof is to apply an implicit function theorem on the dual formulation to get existence of solutions for both on primal and dual problems. We moreover derive an explicit expression of the (conservative) Jacobian of γ.

From constraint optimization theory, the Lagrangian function associated to $(\mathcal{P}_A)$ reads

$$L_z(\lambda, x) \overset{\text{def}}{=} F(x) + \lambda^T \left(A(x - x^\star) - z\right).$$

where $\lambda \in \mathbb{R}^D$ is the vector of Lagrange multipliers and $F(x) = \frac{1}{2}\left\|\Phi(x) - \Phi(x^\star)\right\|_2^2$ is the objective function. As Φ is of class $\mathcal{C}^1$, and given that the constraint is linear, the KKT conditions reduce to

$$\nabla_{L_z}(\lambda, x) = \begin{bmatrix} A(x - x^\star) - z \\ \nabla F(x) + A^T\lambda \end{bmatrix} = 0_{D+N},$$

for $z \in \mathbb{R}^D$. In particular, multiplying the second line of ∇_{L_z} by the matrix A yields that $\lambda(z) = -A\nabla F(x)$ since $AA^T = \mathrm{Id}_D$.

Hence, finding a solution (λ, x) of the KKT conditions of $(\mathcal{P}_A)$ is equivalent to solving the implicit system

$$G(z, x) \overset{\text{def}}{=} \begin{bmatrix} A(x - x^\star) - z \\ (\mathrm{Id}_N - A^T A)\nabla F(x) \end{bmatrix} = 0_{D+N}.$$

Let's notice that G has values in $\mathbb{R}^{D+N}$ but its image spans only the N dimensional subspace $\mathbb{R}^D \times \mathrm{Ker}A$. Indeed, the matrix $(\mathrm{Id}_N - A^T A)$ is the projection onto the subspace $\ker A = (\mathrm{im}A^T)^\perp$. Thus one can reduce the output dimension of G to make it surjective while keeping the same zeros level sets. This is done by left multiplying with $[A^T, \mathrm{Id}_N]$ which embeds the first term $A(x - x^\star) - z$ of G onto $\mathrm{im}A^T$ while keeping the second term in $(\mathrm{im}A^T)^\perp$. One obtains

$$H(z, x) \overset{\text{def}}{=} A^T(A(x - x^\star) - z) + (\mathrm{Id}_N - A^T A)\nabla F(x).$$

Our problem is now equivalent to the following implicit equation

$$H(z, \gamma(z)) = 0_N, \tag{49}$$

for $z \in \mathbb{R}^D$ and with $H : \mathbb{R}^D \times \mathbb{R}^N \to \mathbb{R}^N$ defined before.

Given that a solution of Equation 49 verifies the KKT conditions, it thus is also a solution of the primal equation $(\mathcal{P}_A)$. We now use two forms of implicit function theorem to get existence of a solution of Equation 49 and thus existence of a mapping γ for $(\mathcal{P}_A)$.

- $\mathcal{C}^2$ *case:* From implicit function theorem (Lee, 2012, Thm. 5.15), the only assumption required is the invertibility condition on the second sub-matrix $\mathcal{D}_2 H(0_D, x^\star)$ of the differential $\mathcal{D}H = [\mathcal{D}_1 H, \mathcal{D}_2 H]$ of H at $(0_D, x^\star)$. With some computations, one exactly recovers the second condition in Assumption 19. Then as a consequence, the Jacobian of γ is given by

  $$J_\gamma(z) = -\mathcal{D}_2 H(z, \gamma(z))^{-1} \mathcal{D}_1 H(z, \gamma(z))$$

  from which we recover the Equation 47.

- $\mathcal{C}^1$ *case:* We use the Lipschitz definable version of implicit function theorem from (Bolte et al., 2021, Thm. 1). As J_Φ is locally Lipschitz and definable, so is F and thus H. With $\gamma(0_D) = x^\star$, one recovers $H(0_D, \gamma(0_D)) = 0_N$. A conservative Jacobian of H can be written as

  $$\tilde{J}_H : (z, x) \rightrightarrows \left\{ \left[-A^T, \Pi_{\mathrm{im} A^T} + \Pi_{\ker A} \cdot B \right] \ ; \ B \in \tilde{J}_{\nabla F}(\gamma(z)) \right\}. \tag{50}$$

  Since $\nabla F(x) = J_\Phi(x)^T(\Phi(x) - \Phi(x^\star))$, by path-differentiation rules, one derives the formula

  $$\tilde{J}_{\nabla F}(x) = \left\{ B \ (\Phi(x) - \Phi(x^\star)) + J_\Phi(x)^T J_\Phi(x) \ ; \ B \in \tilde{J}_{J_\Phi^T}(x) \right\}$$

  and evaluating in $x = x^\star$ provides, since the first term vanishes,

  $$\tilde{J}_{\nabla F}(x^\star) = \left\{ J_\Phi(x^\star)^T J_\Phi(x^\star) \right\}. \tag{51}$$

  Finally inserting Equation 51 in Equation 50 allows us to express

  $$\tilde{J}_H(0_D, x^\star) = \left\{ \left[-A^T, \Pi_{\mathrm{im} A^T} + \Pi_{\ker A} \cdot J_\Phi(x^\star)^T J_\Phi(x^\star) \right] \right\}.$$

  This last equation implies that the assumptions of (Bolte et al., 2021, Thm. 1) are verified: $\tilde{J}_H(0_D, x^\star)$ is a singleton and thus convex and its second sub-matrix is invertible from Assumption 19. This completes the proof.

■

A.3.2 Proof of Theorem 8, Theorem 9, Proposition 11 and Theorem 12

Assumption 10, Proposition 11 and Theorem 12 are specific instances of Assumption 22, Proposition 23 and Theorem 24 below with $A = V_D^{\star,T}$.

Assumption 22 (Controlled thickness) *Let*

$$\mathcal{S}_z = \{x \in \mathcal{U}_\delta^\varepsilon, A(x - x^\star) = z\}$$

denote the slice of $\mathcal{U}_\delta^\varepsilon$ which has coordinate z on $\mathcal{T}_{x^\star}$. We assume that

$$\sup_{z:\|\gamma(z)-x^\star\|_2\leq\delta} \operatorname{diam}(\mathcal{S}_z) \leq \eta. \tag{52}$$

Proposition 23 (Uniformly bounded curvature) *Let Φ be of class $\mathcal{C}^1$. Assume that its Jacobian J_Φ is L-Lipschitz over the ball $\mathcal{B}(x^\star, \delta)$, i.e.*

$$\forall x, x' \in \mathcal{B}(x^\star, \delta), \quad \|J_\Phi(x) - J_\Phi(x')\|_{op} \leq L \left\|x - x'\right\|_2. \tag{53}$$

Then assume that the Lipschitz constant L satisfies

$$L < \frac{\sigma_{\min}}{\delta} \tag{54}$$

where $\sigma_{min} > 0$ is the lowest singular value of the Jacobian $J_\Phi(x^\star)$ restricted to the kernel of A. Then Assumption 22 is verified with

$$\eta = \frac{2\varepsilon}{\sigma_{min} - L\delta}. \tag{55}$$

Proof Setting $z \in \mathcal{B}(0_D, \delta)$, let's upper bound the diameter of the intersection of $\mathcal{U}_\delta^\varepsilon$ with the affine hyperplane $A(\cdot - x^\star) = z$. When this slice is empty there is nothing to do, and when it is not, it only remains to bound the norm $\|x_1 - x_2\|_2$ for any couple $x_1, x_2 \in \mathcal{U}_\delta^\varepsilon$ verifying $A(x_1 - x^\star) = A(x_2 - x^\star) = z$.

To that end, one can remark that

$$\|\Phi(x_1) - \Phi(x_2)\|_2 \leq \|\Phi(x_1) - \Phi(x^\star)\|_2 + \|\Phi(x^\star) - \Phi(x_2)\|_2 \leq 2\varepsilon$$

as x_1 and x_2 belong to $\mathcal{U}_\delta^\varepsilon$. The remaining of the proof consists in lower bounding the left-hand side by a term of the form $\mu \|x_1 - x_2\|_2$ with some $\mu > 0$. Applying integral form of Taylor expansion on Φ, one gets

$$\Phi(x_1) - \Phi(x_2) = \bar{J} \cdot (x_1 - x_2), \tag{56}$$

with the operator $\bar{J} \in \mathbb{R}^{M\times N}$ defined as

$$\bar{J} \stackrel{\text{def}}{=} \int_0^1 J_\Phi\left(x_2 + t(x_1 - x_2)\right) \mathrm{d}t. \tag{57}$$

Then, from the inverse triangular inequality, we get

$$\|\Phi(x_1) - \Phi(x_2)\|_2 \geq \|J_\Phi(x^\star)(x_1 - x_2)\|_2 - \left\|\left(\bar{J} - J_\Phi(x^\star)\right)(x_1 - x_2)\right\|_2. \tag{58}$$

The first right term of Equation 58 can be handled noticing that the difference $x_1 - x_2$ lies in the kernel of A since $A(x_1 - x_2) = A(x_1 - x^\star) + A(x^\star - x_2) = z - z = 0$. Thus by definition of the minimal singular value of $J_\Phi(x^\star)$ restricted to the kernel of A, denoted $\sigma_{\min}$, it follows that

$$\|J_\Phi(x^\star)(x_1 - x_2)\|_2 \geq \sigma_{\min} \|x_1 - x_2\|_2 .$$

Moreover, the second right term of Equation 58 can be bounded using Lipschitz constant as follows:

$$\begin{aligned}
\left\| \left(\bar{J} - J_\Phi(x^\star)\right)(x_1 - x_2) \right\|_2 &\leq \int_0^1 \left\| \left[J_\Phi\left(x_2 + t(x_1 - x_2)\right) - J_\Phi(x^\star)\right](x_1 - x_2) \right\|_2 \mathrm{d}t \\
&\leq \int_0^1 \left\| J_\Phi\left(x_2 + t(x_1 - x_2)\right) - J_\Phi(x^\star) \right\|_{\mathrm{op}} \mathrm{d}t \cdot \|x_1 - x_2\|_2 \\
&\leq L \int_0^1 \|x_2 + t(x_1 - x_2) - x^\star\|_2 \, \mathrm{d}t \cdot \|x_1 - x_2\|_2 \\
&\leq L\delta \|x_1 - x_2\|_2 .
\end{aligned}$$

Finally Equation 58 becomes

$$\|\Phi(x_1) - \Phi(x_2)\|_2 \geq (\sigma_{\min} - L\delta) \|x_1 - x_2\|_2$$

as intended. ■

Proof [Proof of Theorem 9] Using a standard result on smooth manifolds from Lee (2012), which is summarized in Proposition 18, the image of γ locally defines an embedded manifold since $J_\gamma(0)$ is injective as the product of an invertible matrix with A^T which is injective by assumption. ■

Proof [Proof of Theorem 8] The goal of this proof is to demonstrate that $\mathcal{U}^0$ is locally the image of a smooth transformation of a D-dimensional affine subspace (step 1), and that this transformation preserves the local bijectivity of the orthogonal projection from the uncertainty region onto its tangent space (step 2). This result follows from an application of the inverse function theorem.

As a consequence, each point in the tangent space near the origin corresponds to a unique point in the uncertainty region. Therefore, the optimization problem defining the Jackpot parametrization admits a unique solution in a sufficiently small neighborhood, and this solution coincides with $\mathcal{U}^0$. A graphical view of the geometric objects used in this proof can be found in Fig. 16.

Step 1: By the constant rank theorem, since the rank of J_Φ is constant in a neighborhood of $x^\star$, there exist neighborhoods $\mathcal{V}$ of $x^\star$ and $\mathcal{W}$ of $\Phi(x^\star)$, and two diffeomorphisms $v : \mathbb{R}^N \to \mathcal{V}$ and $w : \mathbb{R}^M \to \mathcal{W}$ such that

$$\Phi|_{\mathcal{U}} = w \circ J_\Phi(x^\star) \circ v^{-1}. \tag{59}$$

Notice that $\mathcal{U}^0 \cap \mathcal{V} = v(\mathcal{L})$, where

$$\mathcal{L} \stackrel{\text{def}}{=} J_\Phi(x^\star)^{-1} \left(\left\{ w^{-1}(\Phi(x^\star)) \right\} \right) \tag{60}$$

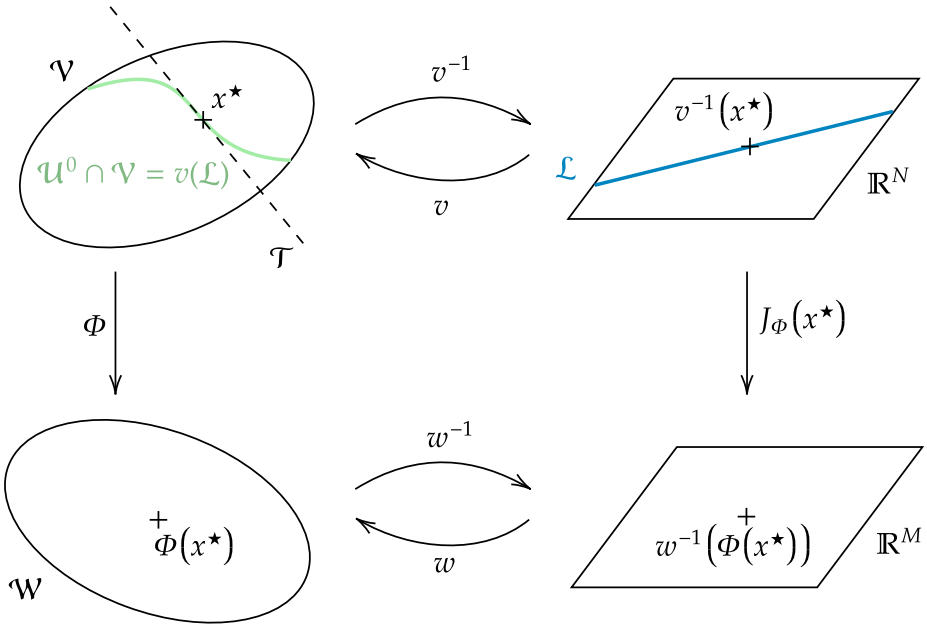


Figure 16: Illustration of the decomposition of the direct model Φ thanks to constant rank theorem.

is an affine subspace of $\mathbb{R}^N$.

Step 2: Let $\mathcal{T}$ be the tangent space of $v(\mathcal{L})$ at $x^\star$ and $\pi_\mathcal{T} : \mathbb{R}^N \to \mathcal{T}$ its orthogonal projection. Let's show that $f \overset{\text{def}}{=} \pi_\mathcal{T} \circ v : \mathcal{L} \to \mathcal{T}$ is locally a diffeomorphism around $x_0 \overset{\text{def}}{=} v^{-1}(x^\star)$ using inverse function theorem. The differential of f is given by

$$D_f(x_0) = D_{\pi_\mathcal{T}}(x^\star) \cdot D_v(x_0) = \pi_\mathcal{T} \cdot D_v(x_0). \tag{61}$$

Moreover, by definition of the tangent space $\mathcal{T}$, one has

$$D_v(x_0)(\mathcal{L}) = \mathcal{T}. \tag{62}$$

Hence $D_f(x_0)|_\mathcal{L} = D_v(x_0)|_\mathcal{L}$. The key argument is that since the linear application $D_v(x_0) : \mathbb{R}^N \to \mathbb{R}^N$ is invertible (because v is a diffeomorphism), the same goes for $D_v(x_0)|_\mathcal{L} : \mathcal{L} \to \mathcal{T}$ since $\mathcal{T}$ and $\mathcal{L}$ have the same dimension. Hence applying inverse function theorem, the map $f : \mathcal{L} \to \mathcal{T}$ is a local diffeomorphism near $v^{-1}(x^\star)$. It induces that $\pi_\mathcal{T}|_{v(\mathcal{L})} = f \circ v^{-1} : v(\mathcal{L}) \to \mathcal{T}$ is also a local diffeomorphism near $x^\star$. Let take $\mathcal{V} = \mathcal{B}(x^\star, \delta)$ with $\delta > 0$ such that $\pi_\mathcal{T}|_{\mathcal{U}^0_\delta}$ is a diffeomorphism to its image. Thus normal slices $x' + \mathcal{T}^\perp$ for $x' \in \mathcal{T} \cap \mathcal{B}(x^\star, \delta)$ intersect $\mathcal{U}^0$ at a unique point. As a consequence, the problem

$$\underset{\pi_\mathcal{T}(x)=z,\ \|x\|_2 \leq \delta}{\operatorname{argmin}} \frac{1}{2} \|\Phi(x) - \Phi(x^\star)\|_2^2 \tag{63}$$

that defines the Jackpot parameterization admits a unique global minimizer and thus $\mathcal{M}^0_\delta = \mathcal{U}^0_\delta$. ■

Theorem 24 *Let $\Phi : \mathbb{R}^N \to \mathbb{R}^M$ be a function of class $\mathcal{C}^2$, $x^\star \in \mathbb{R}^N$ be a point, $A \in \mathbb{R}^{D\times N}$ satisfying Assumption 19, and let $\varepsilon > 0$ be a threshold. Then $\mathcal{M}_\delta^\varepsilon$ defines a manifold with parameterization γ in $(\mathcal{P}_A)$ for sufficiently small $\delta > 0$. Moreover, under Assumption 22, the manifold $\mathcal{M}_\delta^\varepsilon$ is a good approximation of $\mathcal{U}_\delta^\varepsilon$ in the sense that:*

$$d_{\mathcal{H}}(\mathcal{M}_\delta^\varepsilon,\ \mathcal{U}_\delta^\varepsilon) < \eta, \tag{64}$$

where the Hausdorff distance $d_{\mathcal{H}}$ is defined in Equation 2. More precisely, we have the following inclusions:

$$\mathcal{M}_\delta^\varepsilon \subseteq \mathcal{U}_\delta^\varepsilon \subseteq \mathcal{M}_\delta^\varepsilon + \mathcal{B}(0,\eta). \tag{65}$$

Proof First of all, from Theorem 21, the set $\mathcal{M}_\delta^\varepsilon$ is well defined for some $\delta > 0$ and by definition verifies the first inclusion $\mathcal{M}_\delta^\varepsilon \subseteq \mathcal{U}_\delta^\varepsilon$.

To prove the second inclusion, given a point $\tilde{x} \in \mathcal{U}_\delta^\varepsilon$, we need to find a point $x' \in \mathcal{M}_\delta^\varepsilon$ such that $\|\tilde{x} - x'\|_2 \leq \eta$. A natural candidate is $x' \overset{\text{def}}{=} \gamma(\tilde{z})$ with $\tilde{z} \overset{\text{def}}{=} A(\tilde{x} - x^\star)$. Indeed, given that both $\tilde{x}$ and x' are in $\mathcal{U}_\delta^\varepsilon$ and (by definition) verify $A(\tilde{x} - x^\star) = A(x' - x^\star) = \tilde{z}$, Assumption 22 implies that $\|\tilde{x} - x'\|_2 \leq \eta$. Moreover, we have

$$\|\Phi(\gamma(\tilde{z})) - \Phi(x^\star)\|_2 \leq \|\Phi(\tilde{x}) - \Phi(x^\star)\|_2 \leq \varepsilon, \tag{66}$$

and thus $x' = \gamma(\tilde{z}) \in \mathcal{M}_\delta^\varepsilon$. Note that the first inequality in Equation 66 comes from the fact that $\gamma(\tilde{z})$ solves $(\mathcal{P}_A)$ while $\tilde{x}$ is an admissible point. The second one is due to the fact that $\tilde{x} \in \mathcal{U}_\delta^\varepsilon$. This completes the proof. ■

A.4 Numerical Computation

A.4.1 Proof of Proposition 14

Proof Proof of Proposition 14. In $(\mathcal{P}_z)$, splitting the vector $x - x^\star \in \mathbb{R}^N$ within the space decomposition $\mathbb{R}^N = \mathrm{Im}V_D^\star \oplus (\mathrm{Im}V_D^\star)^\perp$ yields the reformulation $(\mathcal{P}'_z)$. Here the constraint $V_D^{\star,T}(x - x^\star) = z$ is included in the objective function that we denote

$$F_z : x \mapsto \frac{1}{2}\|\Phi(r(z,x)) - \Phi(x^\star)\|_2^2 \tag{67}$$

where $r(z,x) \overset{\text{def}}{=} x^\star + V_D^\star z + \Pi_\perp x$.

The main strategy to locally prove the linear convergence of gradient descent is to show a geometrical property on F_z. Since it is not locally strongly convex we use some *Polyak-Lojasiewicz (PL) condition*:

Definition 25 *For a real number $\mu > 0$, a function f is said to verify the* μ-Lojasiewicz inequality *if for all x,*

$$\frac{1}{2}\|\nabla f(x)\|_2^2 \geq \mu(f(x) - f(x^\star)). \tag{68}$$

The argumentation will proceed according to the following four steps, where the results are verified locally around $(z,x) = (0_D, 0_N)$:

1. The restricted map $\tilde{F}_z \overset{\text{def}}{=} F_z|_{(\text{Im}V_D^\star)^\perp}$ on $(\text{Im}V_D^\star)^\perp$ is strongly convex.

2. The map $\tilde{F}_z$ satisfies a PL inequality since any strongly convex map also satisfies a PL inequality (Polyak, 1963, lemma).

3. Since F_z only depends on the orthogonal part $(\text{Im}V_D^\star)^\perp$, i.e. $F_z(x) = F_z(x')$ whenever $\Pi_\perp x = \Pi_\perp x'$, the objective function F_z also verifies the same PL inequality.

4. Since F_z verifies a PL inequality and is also L-smooth for some $L > 0$ (since of class $\mathcal{C}^1$ on a neighbor set), this assures the linear convergence of the gradient descent (Polyak, 1963, thm. 4).

The only remaining argument is to prove the first point : the local strong convexity of the objective function. The Hessian of F_z at x is

$$\begin{aligned}\nabla^2 F_z(x) =& \Pi_\perp J_\Phi^T(r(z,x)) J_\Phi(r(z,x)) \Pi_\perp \\ &+ \Pi_\perp H_\Phi^T(r(z,x))(\Phi(r(z,x)) - \Phi(x^\star)) \in \mathbb{R}^{N\times N}.\end{aligned}$$

Evaluating at $(z,x) = (0_D, 0_N)$ gives

$$H^\star \overset{\text{def}}{=} \Pi_\perp J^{\star,T} J^\star \Pi_\perp \in \mathbb{R}^{N\times N}.$$

Restricted to the subspace $(\text{Im}V_D^\star)^\perp$, this matrix is a positive-definite matrix with eigenvalues equal to $\sigma_1^2 \geq \cdots \geq \sigma_{N-D}^2 > 0$ since $D > R^\star$. Moreover, this submatrix corresponds to the Hessian of the map $\tilde{F}_0$ at $x = 0_N$. By smoothness of Φ, the map $(z,x) \mapsto \nabla^2 \tilde{F}_z(x)$ is continuous and thus the matrices $\nabla^2 \tilde{F}_z(x)$ remain locally positive-definite matrix for (z,x) on a neigborhood $\mathcal{U} \times \mathcal{V}$ of $(0_D, 0_N)$. This property is a characterization of the strong convexity of the maps $\tilde{F}_z$ retricted to $\mathcal{V}$ for all $z \in \mathcal{U}$. ■

A.4.2 Proof of Corollary 15

From Theorem 5, one knows that the parameterization function γ is locally Lipschitz. The key remaining argument is that any point z of the ball $\mathcal{B}_D(0,\delta)$ is at most at a distance of $\sqrt{D}s/2$ from a point z' of the grid Z_s. From Lipschitz consideration, $\|\gamma(z) - \gamma(z')\|_2 \leq L\sqrt{D}s/2$. This leads, using Theorem 24, the following inequalities

$$\begin{aligned}d_{\mathcal{H}}(\mathcal{U}_\delta^\varepsilon,\ \mathcal{M}_\delta^\varepsilon(Z)) &\leq d_{\mathcal{H}}(\mathcal{U}_\delta^\varepsilon,\ \mathcal{M}_\delta^\varepsilon) + d_{\mathcal{H}}(\mathcal{M}_\delta^\varepsilon,\ \mathcal{M}_\delta^\varepsilon(Z)) \\ &\leq \eta + L\sqrt{D}s/2,\end{aligned}$$

which completes the proof.

References

P. A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds.* Princeton University Press, 2008.

V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen. On instabilities of deep learning in image reconstruction and the potential costs of ai. *Proceedings of the National Academy of Sciences*, 117(48):30088–30095, 2020.

R. Arutjunjan, B. M. Schaefer, and C. Kreutz. Constructing exact confidence regions on parameter manifolds of non-linear models. *arXiv preprint arXiv:2211.03421*, 2022.

J. M. Bardsley. MCMC-based image reconstruction with uncertainty quantification. *SIAM Journal on Scientific Computing*, 34(3):A1316–A1332, 2012.

J. M. Bardsley, A. Solonen, H. Haario, and M. Laine. Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1895–A1910, 2014.

D. J. Bates, J. D. Hauenstein, and N. Meshkat. Identifiability and numerical algebraic geometry. *Plos one*, 2019.

G. Bellu, M. P. Saccomani, S. Audoly, and L. D'Angiò. Daisy: A new software tool to test global identifiability of biological and physiological systems. *Computer methods and programs in biomedicine*, 2007.

J. Bolte, T. Le, E. Pauwels, and T. Silveti-Falls. Nonsmooth implicit differentiation for machine-learning and optimization. *Advances in neural information processing systems*, 34:13537–13549, 2021.

J. Bona-Pellissier, F. Malgouyres, and F. Bachoc. Local identifiability of deep ReLU neural networks: the theory. *Advances in Neural Information Processing Systems*, 35:27549–27562, 2022.

J. C. Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, 3rd edition, 2016. ISBN 978-1119121503. doi: 10.1002/9781119121534.

J. Chemseddine, P. Hagemann, C. Wald, and G. Steidl. Conditional Wasserstein distances with applications in Bayesian OT flow matching. *arXiv preprint arXiv:2403.18705*, 2024.

H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

M. Coste. *An introduction to o-minimal geometry*. Istituti editoriali e poligrafici internazionali Pisa, 2000.

J. A. Duersch, M. Shao, C. Yang, and M. Gu. A robust and efficient implementation of LOBPCG. *SIAM Journal on Scientific Computing*, 2018.

M. C. Eisenberg and M. A. L. Hayashi. Determining identifiable parameter combinations using subset profiling. *Mathematical biosciences*, 2014.

J. L. Fernández-Martínez and Z. Fernández-Muñiz. The curse of dimensionality in inverse problems. *Journal of Computational and Applied Mathematics*, 2020.

J. L. Fernández-Martínez, M. Z. Fernández-Muñiz, and M. J Tompkins. On the topography of the cost functional in linear and nonlinear inverse problems. *Geophysics*, 2012.

J. L. Fernández-Martínez, Z. Fernández-Muñiz, J. L. G. Pallero, and L. M. Pedruelo-González. From Bayes to Tarantola: new insights to understand uncertainty in inverse problems. *Journal of Applied Geophysics*, 98:62–72, 2013.

S. M. Fischer and M. A. Lewis. A robust and efficient algorithm to find profile likelihood confidence intervals. *Statistics and Computing*, 2021.

J. W. Goodman. *Introduction to Fourier optics*. Roberts and Company publishers, 2005.

A. Gossard and P. Weiss. Training adaptive reconstruction networks for blind inverse problems. *SIAM Journal on Imaging Sciences*, 17(2):1314–1346, 2024.

A. V. Grayver and A. V. Kuvshinov. Exploring equivalence domain in nonlinear inverse problems using covariance matrix adaption evolution strategy (CMAES) and random sampling. *Geophysical Journal International*, 2016.

P. Grohs, S. Koppensteiner, and M. Rathmair. Phase retrieval: uniqueness and stability. *SIAM Review*, 62(2):301–350, 2020.

J. C. Helton, J. D. Johnson, C. J. Sallaberry, and C. B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering & System Safety*, 2006.

S. Hengl, C. Kreutz, J. Timmer, and T. Maiwald. Data-based identifiability analysis of non-linear dynamical models. *bioinformatics*, 2007.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

E. Hubert. Essential components of an algebraic differential equation. *Journal of symbolic computation*, 1999.

S. Hurault, A. Leclaire, and N. Papadakis. Gradient step denoiser for convergent Plug-and-Play. *arXiv preprint arXiv:2110.03220*, 2021.

A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6, dec 2005.

IMCCE. position ephemeris, 2019. URL `https://ssp.imcce.fr/forms/ephemeris`.

K. Jaganathan, Y. C. Eldar, and B. Hassibi. Phase retrieval: An overview of recent developments. *Optical Compressive Imaging*, pages 279–312, 2016.

C. Jauberthie, N. Verdière, and L. Travé-Massuyès. Fault detection and identification relying on set-membership identifiability. *Annual Reviews in Control*, 2013.

H. K. Johansen. A man/computer interpretation system for resistivity soundings over a horizontally strafified earth. *Geophysical Prospecting*, 1977.

M. Kech and F. Krahmer. Optimal injectivity conditions for bilinear inverse problems with applications to identifiability of deconvolution problems. *SIAM Journal on Applied Algebra and Geometry*, 2017.

A. V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM journal on scientific computing*, 2001.

N. N. Lam, P. D. Docherty, and R. Murray. Practical identifiability of parametrised models: A review of benefits and limitations of various approaches. *Mathematics and Computers in Simulation*, 2022.

R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. Bayesian imaging using Plug & Play priors: when Langevin meets Tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022.

J. M. Lee. *Introduction to Smooth Manifolds*. Springer New York, 2012.

Y. Li, K. Lee, and Y. Bresler. Identifiability in bilinear inverse problems with applications to subspace or sparsity-constrained blind gain and phase calibration. *IEEE Transactions on Information Theory*, 63(2):822–842, 2016.

E. Liu, K. Rivera, and J. Fox. Efficient eigensolvers and their applications. 2020.

B. Merkt, J. Timmer, and D. Kaschek. Higher-order Lie symmetries in identifiability and predictability analysis of dynamic models. *Physical Review E*, 92(1):012920, 2015.

H. Miao, X. Xia, A. S. Perelson, and H. Wu. On identifiability of nonlinear ODE models and applications in viral dynamics. *SIAM review*, 2011.

K. Mosegaard and A. Tarantola. Monte Carlo sampling of solutions to inverse problems. *Journal of Geophysical Research: Solid Earth*, 100(B7):12431–12447, 1995.

M. J. et al. Muckley. Results of the 2020 fastMRI challenge for machine learning MR image reconstruction. *IEEE Transactions on Medical Imaging*, 2021. doi: 10.1109/TMI.2021.3075856.

S. Nah, S. Son, S. Lee, R. Timofte, and K. M. Lee. NTIRE 2021 challenge on image deblurring, 2021. URL `https://arxiv.org/abs/2104.14854`.

Beresford N Parlett. *The symmetric eigenvalue problem*. SIAM, 1998.

M. Phuong and C. H. Lampert. Functional vs. parametric equivalence of ReLU networks. In *International Conference on Learning Representations*, 2019.

A. Pinkus. *N-widths in Approximation Theory*, volume 7. Springer Science & Business Media, 2012.

B. T. Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.

B. T. Polyak. Introduction to optimization. 1987.

D. V. Raman, J. Anderson, and A. Papachristodoulou. Delineating parameter unidentifiabilities in complex models. *Physical Review E*, 2017.

A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 2009.

W. C. Rheinboldt. Manpak: A set of algorithms for computations on implicitly defined manifolds. *Computers & Mathematics with Applications*, 1996.

T. J. Rothenberg. Identification in parametric models. *Econometrica: Journal of the Econometric Society*, 1971.

E. Y. Sidky and X. Pan. Report on the AAPM deep-learning spectral CT grand challenge. *Medical physics*, 2024. URL `https://arxiv.org/abs/2212.06718`.

G. W. Stewart and J. G. Sun. *Matrix Perturbation Theory*. Academic Press, Boston, 1990. ISBN 978-0126702309.

J. Tachella, M. Terris, S. Hurault, A. Wang, D. Chen, M. H. Nguyen, M. Song, T. Davies, L. Davy, J. Dong, et al. Deepinverse: A python package for solving imaging inverse problems with deep learning. *arXiv preprint arXiv:2505.20160*, 2025.

D. Y. W. Thong, C. K. Mbakam, and M. Pereyra. Do Bayesian imaging methods report trustworthy probabilities? *arXiv preprint arXiv:2405.08179*, 2024.

M. K. Transtrum and P. Qiu. Model reduction by manifold boundaries. *Physical review letters*, 2014.

L. Van den Dries. *Tame topology and o-minimal structures*, volume 248. Cambridge university press, 1998.

D. J. Venzon and S. H. Moolgavkar. A method for computing profile-likelihood-based confidence intervals. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 37(1):87–94, 1988.

A. F. Villaverde, N. D. Evans, M. J. Chappell, and J. R. Banga. Input-dependent structural identifiability of nonlinear systems. *IEEE Control Systems Letters*, 3(2):272–277, 2018.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural Comput.*, 23:1661–1674, jul 2011. URL `https://doi.org/10.1162/NECO_a_00142`.

F.-G. Wieland, A. L. Hauber, M. Rosenblatt, C. Tönsing, and J. Timmer. On structural and practical identifiability. *Current Opinion in Systems Biology*, 2021.

K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6360–6376, 2021.