Highlights

Preserving positivity of Gauss-Newton Hessian through random sampling

Kathrin Hellmuth, Christian Klingenberg, Qin Li

- A novel perspective is proposed to consider experimental design as a down-sampling problem from a comprehensive set of design variables.
- In a least squares optimization framework, the problem of preserving local convexity of the loss function is formulated into the positivity-preserving problem of the Gauss-Newton Hessian. Matrix sketching techniques are then deployed to provide quantitative bounds for down-sampling.
- To execute the down-sampling, gradient-free ensemble sampling methods are deployed to select samples.
- Numerical validation is provided using the Schrödinger potential reconstruction problem as an example, where the optimal sensor placements problem is investigated.

Preserving positivity of Gauss-Newton Hessian through random sampling

Kathrin Hellmuth[a], Christian Klingenberg[a], Qin Li[b,]

[a]*Department of Mathematics, University of Würzburg, Emil-Fischer-Straße 40, Würzburg, 97074, BY, Germany*
[b]*Department of Mathematics, University of Wisconsin-Madison, 719 Van Vleck Hall, 480 Lincoln Dr, Madison, 53706, WI, USA*

Abstract

Numerically the reconstructability of unknown parameters in inverse problems heavily relies on the chosen data. Therefore, it is crucial to design an experiment that yields data that is sensitive to the parameters. We approach this problem from the perspective of a least squares optimization, and examine the positivity of the Gauss-Newton Hessian at the global minimum point of the objective function. We propose a general framework that provides an efficient down-sampling strategy that can select data that preserves the strict positivity of the Hessian. Matrix sketching techniques from randomized linear algebra is heavily leaned on to achieve this goal. The method requires drawing samples from a certain distribution, and gradient free sampling methods are integrated to execute the data selection. Numerical experiments demonstrate the effectiveness of this method in selecting sensor locations for Schrödinger potential reconstruction.

Keywords: positivity preserving, inverse problems, Randomized Numerical Linear Algebra, Sampling methods, Schrödinger potential reconstruction,

*Corresponding author
Email addresses: kathrin.hellmuth@uni-wuerzburg.de (Kathrin Hellmuth), klingen@mathematik.uni-wuerzburg.de (Christian Klingenberg), qinli@math.wisc.edu (Qin Li)
URL: https://www.mathematik.uni-wuerzburg.de/en/fluidmechanics/team/kathrin-hellmuth/ (Kathrin Hellmuth), https://www.mathematik.uni-wuerzburg.de/en/fluidmechanics/team/christian-klingenberg/ (Christian Klingenberg), https://math.wisc.edu/~qinli (Qin Li)

Ensemble methods
2000 MSC: 49N45, 65N21, 65Fxx, 62Dxx, 49M41, 90C31,

---

1. Introduction

Inverse problems are ubiquitous. A system in the forward setting maps the parameter to data:

$$y(u) = F(u;p) + \quad ; \tag{1}$$

where F is the map, p is the parameter, and stands for measurement error.

$$F : \quad \Omega \quad \mathsf{R}^K \quad ! \ \mathsf{R}; \quad \text{with} \quad \Omega \quad \mathsf{R}^{d_u} ; \quad \text{and} \quad j\Omega j \quad 1 : \tag{2}$$

When the parameter p is fixed, the forward problem returns the solution $y(u)$ for every $u \ 2 \ \Omega$, the design variable set. $j\Omega j$ denotes the number of possible experiments/readable data, and it can be infinity.

The associated inverse problem is to revert the process: given the reading of y, we are to infer parameter p. There are many approaches to execute this inversion, and optimization is one popular choice:

$$p \ = \operatorname{argmin}_p C(p) = ky(\ ) \quad F(\ ;p)k^2 : \tag{3}$$

Often there are abundant choices in the design set, namely $j\Omega j \quad K$. In this case, it is natural to suspect that one does not need the full data set of $fy(u)g_{u2}$, which contains much more data than necessary. The task at hand is to select a down-sampled y that can give an almost equally good recovery of p. This reduces experimental as well as computational cost, and sometimes renders the problem computationally or experimentally tractable at all [1]. More specifically, we are to design a subset Ω_c, either through a deterministic or random selection process, and the down-sampled data:

$$j\Omega_c j = c \quad j\Omega j ; \quad \text{and accordingly define} \quad y_c = yj \ _c ; \quad F_c = Fj \ _c ;$$

so that

$$p \ \underset{\text{hopefully}}{|\{z\}} \ p_{c;} \ = \operatorname{argmin}_p C_c(p) = ky_c(\ ) \quad F_c(\ ;p)k^2 ; \tag{4}$$

and thus recovering (3) using a smaller set of data.

There are many perspectives to take to compare (3) and (4). In the linear setting when F becomes a matrix $F(u;p) = \mathsf{A}_{u;:} \ p$, the optimization

problem (3) is quadratically convex. The solution is explicit and the Hessian of the landscape is independent of p: $\mathsf{A}^\top\mathsf{A}$. In this setting, the problem of reducing (3) to (4) resembles optimal design, sometimes referred to optimal experimental design, with the goal being finding the rows of A that are "informative." This task is usually translated to analyzing the spectrum of $\mathsf{A}_{\Omega_c}^\top \mathsf{A}_{\Omega_c}$, with A_{Ω_c} standing for down-sampling A by only keeping the rows whose indices are in Ω_c. This matrix can be viewed as a down-sampled Hessian. The goal is to design Ω_c well so that this down-sampled Hessian has a good conditioning. The standard quantities to consider are its trace (A-optimal) and determinant (D-optimal) [2, 3, 4, 5, 6, 7]. We also refer readers to a very nice review in [8].

In the nonlinear setting, for any given general F, this nice structure is lost, and the objective function usually becomes non-convex, with Hessian depending on p. Studying the whole landscape is infeasible, but we can nevertheless examine the local behavior of the objective function around the global minimum point. In particular, if the objective function in (3) is strictly convex close to the global minimum, we hope the down-sampled problem (4) is also, with a preserved convexity constant. This poses the major question we are to address in this paper

How to down-sample data y to ensure local strong convexity of (4)?

This change of perspective from global to local gives us some freedom to address the problem in a general setting. The goal of the current work is to spell out a generic condition for c, and a generic down-sample strategy that still achieves the convexity for a very general class of problems. The proposed sampling strategy is probabilistic in nature, and thus the strong convexity with a preserved convexity coefficient can only be guaranteed with a high probability. This sampling strategy, when applied to any specific problem, leads to a specific distribution for constructing the mask Ω_c. This distribution incorporates the property of F, and thus integrates the knowledge from the underlying model.

The technical preparation of our approach comes from a seemingly unrelated research area of randomized linear algebra (RNLA) [9]. Indeed, the strong convexity of an objective function is coded in its Hessian matrix, and in the vicinity of the global minimum, linearization is a very good approximation, and the Hessian of (3) enjoys a special tensor structure (that sometimes termed Gauss-Newton Hessian [10, 11, 12, 13]). This special structure allows us to deploy random sketching techniques from RNLA, to pin the conditions

for ensuring the positivity. Specifically in this context, we can spell out a probability distribution to draw Ω_c, and show that with high probability, the associated down-sampled Hessian is strictly positive, and thus Problem (4) still enjoys the local strong convexity.

The integration of probabilistic methods to design tasks is currently at its fancy and has been studied for instance in [14, 15] for with matrix sketching techniques for the input-to output map or a low rank basis representation of the data, respectively. In a Bayesian optimal design setting, a data and model adapted random mask for MRI data acquisition could be constructed in [16].

The two main technical pillars of our proposed method is the matrix sketching, and probability sampling method. We briefly review them in Section 2.1 and Section 2.2 respectively. In Section 3 we turn back to the problems (3)-(4), and examine their Hessian's relation around the global minimum. The problem will be cast in a setting to invite the direct use of random sketching. Such application to our context is discussed in Section 3.2 that will lead to a very concrete down-sample strategy. Theoretical guarantees will be provided also in this section. To execute this strategy, practical considerations about sampling choices also play a vital role, and they are discussed in Section 3.3. In Section 4, we apply this general program to the potential reconstruction problem for the Schrödinger equation, and we conclude the article in Section 5.

2. Preview of technical preparations

Two main bodies of technical preparation for the current work are matrix sketching techniques rooted in randomized numerical linear algebra (RNLA), and sampling, rooted in Bayesian problems. The material in this section serves as an overview of these tools, and we also unify notations.

2.1. Matrix Sketching by RNLA

RNLA sees its biggest impact in big data applications, where large data sets, that usually exceed RAM capacities, need to be stored and analyzed quickly. Techniques developed within the domain of RNLA typically target at accessing and assessing a subset of data that is reduced in size but still representative, through "sketching", see [9, 17, 18, 19] and references therein.

The technique most relevant to our context is the simple computation of matrix-matrix product: how to compute $\mathsf{A}\mathsf{A}^\top$ efficiently? Of specific inter-

ests, we are now in the regime where $\mathsf{A} \in \mathbb{R}^{K \times N}$ is a very short but fat matrix with $N \gg K$, so it has significantly many more columns than its number of rows. Denoting $\mathsf{A}_{:,i}$ its i-th column, the problem is to compute a rather small-sized matrix B from a large-sized matrix A:

$$\mathbb{R}^{K \times K} \ni \mathsf{B} = \mathsf{A}\mathsf{A}^{\top} = \sum_{i=1}^{N} \mathsf{A}_{:,i}\mathsf{A}_{:,i}^{\top}.$$

The grand philosophy of Monte Carlo is that anything written in a summation form can be interpreted as taking an expectation, and thus can be turned into a sampling problem. In this setting, define the random object

$$\mathsf{X} = \frac{1}{p_i}\mathsf{A}_{:,i}\mathsf{A}_{:,i}^{\top}; \quad \text{with probability } p_i; \quad \text{then} \quad \mathsf{B} = \mathbb{E}(\mathsf{X});$$

and the law of large number indicates:

$$\mathsf{B} \approx \frac{1}{c}\sum_{j=1}^{c} \mathsf{X}_j; \quad \text{where } \mathsf{X}_j \text{ is a draw.} \tag{5}$$

The following algorithm summarizes this proposal:

Algorithm 1 BasicMatrixMultiplication; [9, Algorithm 3]

Input: Matrix $\mathsf{A} \in \mathbb{R}^{K \times N}$, a sample size $c \ll N$ and probabilities $\{p_n\}_{n=1}^{N}$.
Output: Matrix $\mathsf{C} \in \mathbb{R}^{K \times c}$ such that $\mathsf{C}\mathsf{C}^{\top} \approx \mathsf{A}\mathsf{A}^{\top}$.

1: **for** $j = 1, \ldots, c$ **do**
2: Sample n_j i.i.d. using $\mathbb{P}(n_j = m) = p_m$;
3: Set the j-th column of C as $\mathsf{C}_{:j} = \mathsf{A}_{:n_j}/\sqrt{c\,p_{n_j}}$.
4: **end for**
5: **return** C and $\mathsf{C}\mathsf{C}^{\top}$.

Clearly, this algorithm is arrived simply by setting $\mathsf{X}_j = \mathsf{C}_{:j}\mathsf{C}_{:j}^{\top}$ in (5). To justify the algorithm, we need to make the approximation sign in (5) more precise, and spell out the dependence on c and $\{p_i\}$ explicitly, largely by deploying central limit theorem and various application of Chernoff bound. It is worth noting that the random variable here X is a matrix instead of a scalar, so the application of concentration inequality needs caution. Nevertheless, we have a theorem:

Theorem 1 ([9, Theorem 7]). *Let* $\mathsf{A} \in \mathbb{R}^{K \times N}$; $c \ll N$ *and probabilities* $\{p_n\}$, *such that* $\sum_{n=1}^{N} p_n = 1$. *Denote* ε *a positive number* $0 < \varepsilon < 1$ *so that*

$$p_n \geq \beta \frac{\|\mathsf{A}_{:,n}\|_2^2}{\|\mathsf{A}\|_F^2},$$

and let C *be constructed by Algorithm 1, then,* $\mathsf{C}\mathsf{C}^\top$ *approximates* $\mathsf{A}\mathsf{A}^\top$ *with high precision and high probability:*

$$\mathbb{P}\left\{\|\mathsf{A}\mathsf{A}^\top - \mathsf{C}\mathsf{C}^\top\|_F \leq \frac{1 + \sqrt{\frac{8\log(2/\delta)}{\beta c}}}{\beta c}\|\mathsf{A}\|_F^2\right\} \geq 1 - \delta. \tag{6}$$

Here δ *is any prescribed failure rate, and* $\mathbb{P}$ *is taken over all drawings of* C.

The implication of the theorem is the following. If the columns of A are chosen proportional to its "volume" – the L_2 norm of the column – then with high probability $(1 - \delta)$, the approximated B using $\mathsf{C}\mathsf{C}^\top$ is accurate, with the error of the Frobenius norm decaying in the format of $\frac{\beta^{-1}\sqrt{j \log j}}{\beta c}$, where c is the chosen number of columns. This result is roughly expected through the central limit theorem argument.

The optimal choice of the sampling strategy is to set $p_n = \frac{\|\mathsf{A}_{:,n}\|_2^2}{\|\mathsf{A}\|_F^2}$. When this happens $\beta = 1$, and the error term in (6) achieves its minimum. Suppose we set $\delta = 0.01$, then noting $\log(\delta^{-1})$ only gives 2 and is an $O(1)$ number, having the error to be $\varepsilon\|\mathsf{A}\|_F^2$ requires $c \sim \frac{O(1)}{\varepsilon^2}$.

We should stress the statement of the problem does not have explicit dependence on N. Indeed, N can be infinity, and A is a continuously indexed matrix. In our setting, this corresponds to the case when our design space $|\Omega| = \infty$.

2.2. Sampling Algorithms

Sampling is the class of tasks aimed at drawing representative samples from a desired distribution, denoted by π through this section. It often raise in the context of Bayesian sampling, where the target desired distribution is the posterior distribution $\pi(u) \doteq p_{\text{pos}}(u|y) \propto p_{\text{pr}}(u)l(y|u)$. In general, due to the positivity of a probability measure, we denote the target distribution

$$\pi(u) \propto e^{-\Phi(u)}. \tag{7}$$

where Φ is sometimes referred to as the potential, and $\propto$ means that is normalized to be integrable to 1.

Classical methods are predominantly Markov Chain Monte Carlo (MCMC) type algorithms, which corresponds to designing a Markov chain whose invariant measure is the desired target distribution. When a sample walks through this Markov chain, in time, the distribution of the sample converges to the target distribution. Most well-known examples include Langevin Monte Carlo, Hamiltonian Monte Carlo, and Metropolis-Hasting LMC, and so on [20, 21, 22, 23, 24, 25, 26].

Another sampling paradigm that recently has garnered a lot of research interests is the ensemble type method. Originating from data assimilation [27, 28], the idea is then integrated to solve sampling problems, with the well-known examples being Ensemble Kalman Sampler (EKS) [29] or Consensus Based Sampler (CBS) [30]. The idea is to evolve a full set of samples altogether in an interactive manner. The interaction encodes the communication, and is designed to achieve certain properties, such as gradient free and affine invariance. This is an active area of research, and the non-asymptotic convergence theory is yet to be developed.

In our setting, we could choose the sampling methods at our will, so both the classical MCMC and the newly made available ensemble methods can potentially be useful. Since we are choosing a subset of samples $u \in \Omega_c$, evolving the whole set is directly relevant. We discuss EKS and CBS below.

EKS Sampling. EKS can be viewed as an ensemble version of the Langevin dynamics. It allocates computational resources to update N samples of $\{u_n\}_{n=1}^N$ simultaneously:

$$\mathrm{d}u_n = \quad C(U)\, r\, \Phi(u_n)\, \mathrm{d}t + \sqrt{2C(U)}\, \mathrm{d}W_n\,; \tag{8}$$

where $C(U) = N^{-1} \sum_n (u_n - \bar{u}) \otimes (u_n - \bar{u})$ is the empirical covariance matrix between the particles, and $\bar{u} = N^{-1} \sum_{n'} u_{n'}$ is the mean. W_n are independent and identically distributed Brownian motions. Often Φ takes on a quadratic form: $\Phi(u) = \frac{1}{2}\|f(u) - d\|^2$, then if f is mildly nonlinear

$$\begin{aligned} C(U)\, r\, \Phi(u_n) = & \frac{f(u_n) - d}{N} \sum_{n'} (u_{n'} - \bar{u}) \otimes (u_{n'} - \bar{u})\, r\, f(u_n) \\ & \frac{f(u_n) - d}{N} \sum_{n'} (u_{n'} - \bar{u}) \otimes (f(u_{n'}) - \bar{f})\,; \end{aligned} \tag{9}$$

where we used the mild nonlinearity and assumed $r\, f(u_n)$ stays close to a constant for all u_n. The notation $\bar{f} = \frac{\sum_n f(u_n)}{N}$ is used. Though strong assumptions are made, the implementation of (8) is made gradient free, a desired property.

When Φ is Lipschitz-smooth, [31] and [32] showed the mean-field limit of (8) is:

$$@_t \quad + r \quad (C(\ )\, r\, \Phi) = C(\ )\Delta \quad :$$

It is a straightforward computation that $\quad / e \quad$ is an invariant measure. When Φ is strongly convex, it was also shown in [29] that this PDE converges exponentially fast.

In summary, denoting $\quad_N = \frac{1}{N} \sum \quad u_n$ the empirical distribution, for large enough N and t, $\quad_N \quad$, and $fu_n g$ are regarded as samples drawn from the target distribution $\quad / e \quad$.

CBS Sampling. CBS was introduced in [30] as another method to draw a set of samples simultaneously from a target distribution. It relies on the Laplace principle [33]. A set of N particles $fu_n g_{n=1}^N$ evolve according to

$$\mathrm{d}u_n = \quad (u_n \quad M \ (\ _t^N))\, \mathrm{d}t + \quad \sqrt{2(1+\ )\Gamma\ (\ _t^N)}\, \mathrm{d}W_n\,; \tag{10}$$

where $\quad_t^N = \frac{1}{N} \sum_{n=1}^N \quad u_n(t)$ is the empirical distribution. $M \ (\ )$ is the weighted mean parameterized by $\quad : M \ (\ ) := M(L \quad) = \int u\, (L \quad)(\mathrm{d}u)$ with $L \quad = \frac{e}{\int e \quad \mathrm{d}u}$ being the weighted version of $\quad$ and M operator takes the mean of a probability distribution. In the $\quad ! \ 1$ limit, $L \quad$ converges to a Dirac delta centered on the global minimum of Φ over the support of $\quad$, and thus $M \ (\ ) \ ! \ \mathrm{argmin}_u \ \Phi j_{\mathrm{supp}(\ )}$. The second term introduces stochastic deviations in proportion to the covariance of the weighted distribution

$$\Gamma\ (\ ) := \Gamma(L \quad) := \int \quad (u \quad M(L \quad)) \quad (u \quad M(L \quad))\ (L \quad)(\mathrm{d}u)$$

and allows exploration of the distribution landscape. In the mean field limit $N \ ! \ 1$, the particle distribution follows

$$@_t \ = r \quad ((u \quad M(L \quad)) \quad + (1+\ )\Gamma\ (\ )\, r \ ) :$$

Under certain conditions [30], one can show the steady state of this equation is a Gaussian approximation of the target distribution around its global maximum, and the PDE solution converges to it exponentially fast. Furthermore, in [34] the author links this process with Langevin dynamics, viewing it as a gradient-free relaxation.

Greedy Sampling. All the sampling strategies above can be improved. In the MCMC framework, for example, MCMC solvers can be paired-up with a selection process. Sample proposals can be either accepted or rejected according to a certain criteria. A classical example is to introduce Metropolis–Hastings (MH) algorithm, as a post-processing to select "good" samples. This additional effort is minimum, but can de-bias the numerical error introduced in the MCMC step.

Similar strategy can be deployed for ensemble type methods as well. Samples proposed by the algorithm can either be accepted or rejected, depending on a preset criterion. In comparison, only limited work is available in this framework: MH algorithm was introduced to turn biased samples to unbiased ones [35]. We should mention the introduction of MH is mainly to correct the bias. Other criteria can also be introduced that are more specific to the problem at hand. For example, in our case, we examine the convexity of the down-sampled Hessian, so samples are kept or rejected based on if the convexity is improved. This simple strategy is summarized in Algorithm 2 and provokes the ensemble evolution at favourable configurations.

Algorithm 2 Greedy Sampling

Input: initial sample $\{m_j\}_{j=1,\dots,c}$, sample update rule $R : \{1,\dots,N\}^c \to \{1,\dots,N\}^c$, number of iterations $I > 0$, a quantity of interest to be maximized Q

Output: updated sample $\{m_j\}_{j=1,\dots,c}$ with improved evaluation criterion.

1: **for** $i = 1,\dots,I$ **do**
2: Generate sample update: $\{m'_j\}_{j=1,\dots,c} = R(\{m_j\}_{j=1,\dots,c})$:
3: **if** $Q(\{m'_j\}) > Q(\{m_j\})$, **then** Update $\{m_j\}_{j=1,\dots,c} = \{m'_j\}_{j=1,\dots,c}$
4: **end if**
5: **end for**
6: **return** sample $\{m_j\}_{j=1,\dots,c}$.

3. The general program

We now deploy the techniques to solve our experimental design problem. To be more specific, we are tasked to find suitable experimental setups to ensure the local strong convexity of (3), and the global basin preserves positivity when data is down-sampled (4). We will do so by formulating the problem as sketching the Hessian matrix, for which Theorem 1 becomes

handy in providing a theoretical guarantee. This reformulated problem, to be executed in an algorithm, needs to be combined with a sampling strategy, and ensemble type sampling methods, such as (8) and (10) are deployed to serve the purpose.

3.1. Hessian structure at the global basin

Recalling (3) is a non-convex optimization, precise characterization of the landscape for the general F can hardly be made. When we confine ourselves to the vicinity of the global minimum, we can potentially study the local convexity of the problem. This usually translates to finding conditions for achieving the positivity of the Hessian term. In the generic form of (3), Hessian can be computed explicitly.

$$\begin{aligned} H_p C(p) &= \int_\Omega [\, r_p F(u;p) \quad r_p F(u;p) + (F(u;p) \quad y(u)) H_p F(u;p)]\, du \quad (11) \\ &= \underbrace{\int_\Omega G(u;p) G^{>}(u;p)\, du}_{=: H_p C(u;p)} + \int_\Omega (F_n(u;p) \quad y(u)) H_p F(u;p) du; \end{aligned}$$

where $G(u;p)$ collects the gradients, for every fixed u:

$$G(u;p) = r_p F(u;p) \; 2 \; \mathbb{R}^{K \quad 1}:$$

It is clear that there are two terms in the formulation of (11). The second term reflects the perturbation from the groundtruth and is regarded small in this neighborhood. The first has the nice matrix-matrix product form as was discussed in Section 2.1. To proceed we now make three assumptions:

(A1) There is an underlying ground truth parameter $p_?$ such that $y = F(p_?)$.

(A2) The Hessian $H_p C(u;p)$ is uniform continuous in the small neighborhood of $p_?$ for all $u \; 2 \; \Omega$.

(A3) The Hessian at the global optimizer $H_p C(p_?)$ is positive definite.

All these assumptions are valid for a large class of examples. In particular, (A1) states that the measurements are generated by the true model and are pollution free. The continuity requirement in (A2) is also rather mild, and is usually fulfilled by inheriting the parameter smooth-dependence from the inverse problem at hand. For example, in the setting of the PDE-constrained

inversion, this property can frequently be established, as well-posedness of the forward partial differential equation (PDE) typically involves continuous dependence on parameters and initial data, see one example from the authors in [36], where a parameter identification problem for the kinetic chemotaxis equation was studied. (A3) is imposed to prevent the worst case scenario: the problem ought to be strictly convex at the global optimum at least when all data gets used. The failure of this assumption means the Hessian is not strictly positive even when we deploy the information from all available data, suggesting the problem is intrinsically ill.

With all assumptions made, it is straightforward to see:

$$H_p C(p_?) = \int G_?(u) G_?^{>}(u) du \quad 0; \qquad \text{where } G_?(u) := G(u; p_?): \tag{12}$$

This formulation sets us up squarely in the framework specified in Section 2.1. Preserving the positivity with down-sampled data now translates to selecting columns in $G_?$ so that

$$\int G_?(u) G_?^{>}(u) du \quad \frac{1}{c} \sum_{u2 \ c} G_?(u) G_?^{>}(u) \tag{13}$$

with a high probability. Noting that columns in $G_?$ are exactly $r F$, the problem is equivalent to finding the subset of F, so that this subset of experiments can already produce a convex objective function.

Remark 1. *We formulate the problem by only comparing the output of the experiment with the data y. When regularization presents, the cost function becomes* $C_R(p) = C(p) + R(p)$ *for some* > 0 *and a regularization R. Similar results can be established with small modi cations.*

3.2. Experimental Design through Sampling

Following (12) and (13), and combine with Algorithm 1, it is straightforward to arrive at the following. Let the subsampled cost function be:

$$C_c(p) = \frac{1}{2c} \sum_{u2 \ c} \frac{1}{\frac{2}{u}} jF(u; p) \quad y(u)j^2; \tag{14}$$

with Ω_c being a c-subset of Ω. Each $u \ 2 \ \Omega_c$ is an i.i.d. sample from Ω according to the distribution $f \ (u)g$. The associated Hessian $H_p C_c$ at the

global optimizer becomes:

$$H_p C_c(p_?) = \sum_{u2 \ c} \frac{1}{c \ (u)} G_?(u) \quad G_?(u):$$

Local strong convexity of this cost function is assured by the Theorem 2, a quick corollary of Theorem 1, with high probability, given that enough samples were drawn.

Theorem 2. *Consider an inverse problem that satis es assumptions (A1){ (A3) and let the weighted cost function $C_c(p)$ be constructed as in* (14)*, where the sampling distribution* (u) *2* $P(\Omega)$ *satis es*

$$\tilde{} \qquad \text{with} \qquad \tilde{}(u) \ / \ kG_?(u)k_2^2; \tag{15}$$

for some 2 $(0; 1]$*. Furthermore, assume that $kG_?(u)k_2$ is bounded for every u 2* Ω*, then with a su ciently large c, C_c is locally strongly convex at $p_?$ with a high probability.*

To be more precise, for any failure probability 2 $(0; 1)$ *and any minimum value " 2* $(0; \ _{\min}(H_p C(p_?))$*, a choice of the sample size*

$$c \quad kG_?k_F^4 \frac{(1 + \sqrt{8 \quad ^1 \log(\quad ^1)})^2}{(\ _{\min}(H_p C(p_?)) \quad ")^2} \tag{16}$$

assures that with probability at least $1 \quad$ *, the quadratic cost function C_c is locally strongly convex at the true parameter $p_?$ with minimum eigenvalue* $_{\min}(H_p C_c(p_?)) \quad " > 0$*.*

Proof. Assumption (A2) states that the Hessian in continuous in the neighborhood of $p_?$. This implies that to achieve local strong convexity, it is sufficient to show $H_p C_c(p_?) \quad 0$. Noting that

$$\begin{aligned} _{\min}(H_p C_c(p_?)) \quad _{\min}(H_p C(p_?)) \quad & j \ _{\min}(H_p C(p_?)) \quad _{\min}(H_p C_c(p_?))j \\ & _{\min}(H_p C(p_?)) \quad kH_p C(p_?) \quad H_p C_c(p_?)k_F \ ; \end{aligned} \tag{17}$$

and the positivity of $H_p C(p_?) \quad 0$ from Assumption (A3), we are to bound the second term. Using Theorem 1, it is straightforward to see that with probability at least 1

$$kH_p C(p_?) \quad H_p C_c(p_?)k_F = kG_? G_?^T \quad CC^T k_F \\ \frac{1 + \sqrt{8 \quad ^1 \log(\quad ^1)}}{\sqrt{c}} kG_? k_F^2:$$

To achieve $\min(H_p C_c(p_?))$ ", according to (17), we need the term above bounded above by $\min(H_p C(p_?))$ ". The choice of c is given by calling Theorem 1. □

3.3. Practical considerations

According to Theorem 2, we are looking for c i.i.d samples from the optimal probability distribution ˜. As a probability measure over the design space Ω, it can be characterized as:

$$\tilde{}(u) = \frac{1}{Z} e^{(u)} \quad \text{with} \quad \Phi(u) := \log(kG_?(u)k_2^2): \tag{18}$$

A natural application of EKS provides us the following sampling strategy. Set c interactive samples $U = fu_i g_{i=1;:::;c}$ uniformly at initial time, and we evolve them according to

$$\mathrm{d}u_n = \sum_{n^0} D_{n;n^0} u_{n^0}\,\mathrm{d}t + \sqrt{2C(U)}\,\mathrm{d}W_n;$$

where the first term contains the approximation to $C(U)\ r_u\Phi(u_n)$

$$\begin{aligned} C(U)\ r_u\Phi(u_n) &= \frac{1}{N}\sum_{n^0}(u_{n^0}\ \bar{u})\ (u_{n^0}\ \bar{u})\ r_u\Phi(u^n)\Big) \\ &= \frac{2}{NkG_?(u_n)k_2^2}\sum_{n^0}(D_u G_?(u_n)(u_{n^0}\ \bar{u}))^T G_?(u_n)(u_{n^0}\ \bar{u}) \\ &\quad \frac{2}{NkG_?(u_n)k_2^2}\sum_{n^0} G_?(u_{n^0})\ \overline{G_?(U)}^T G_?(u_n)u_{n^0} \\ &=: \sum_{n^0} D_{n;n^0} u_{n^0}; \end{aligned}$$

where we approximated the gradient term by a difference in analogy to (9), with $\overline{G_?(U)} = \frac{1}{N}\sum_n G_?(u_n)$, and used the fact that the $\bar{u}$ term vanishes. Running this SDE forward in time using the classical Euler-Maruyama method gives:

$$u_n^{t_{k+1}} = u_n^{t_k} + \Delta t_k \sum_{n^0} D_{n;n^0}^{t_k} u_{n^0}^{t_k} + \sqrt{2\Delta t_k C(U^{t_k})}\ {}^{t_k}_{n};$$

with $\xi_n^{t_k} \sim N(0; I)$ independent and identically distributed and adaptive time step $\Delta t_k = \frac{t_0}{kD^{t_k}k_F + \varepsilon}$ in dependence of the difference matrix $D^{t_k} = (D^{t_k}_{n;n'})_{n;n'}$ for some $\varepsilon > 0$ as proposed in [37, 29].

Application of CBS is straightforward. As in [30] we deploy the forward in time discretization using an exponential integrator:

$$u_n^{t_{k+1}} = e^{-\gamma t} u_n^{t_k} + (1 - e^{-\gamma t}) M^{-\frac{1}{2}} \left(\xi_{t_k}^N\right) + \sqrt{\frac{(1 - e^{-2\gamma t})(1+\gamma)}{\gamma}} \Gamma^{-\frac{1}{2}} \left(\xi_{t_k}^N\right)_n^{t_k};$$

Remark 2. *We make a remark on the accuracy of these algorithms. One key drawback of ensemble based method is the lack of non-asymptotic convergence rate. The samples provided by these methods are not necessarily the best samples drawn from the optimal distribution. Meanwhile, though the bound in Theorem 2 is explicit, the constants depend on quantities are not known a-priori (e.g. either* $\|G_\dagger\|_F^2 = \int_X \|G_\dagger(u)\|_2^2 du$ *or the minimum eigenvalue* $\lambda_{\min}(H_p C(p_\dagger))$ *are known a-priori), bringing another uncertainty to set parameters.*

However, we should note that drawing samples from the target distribution is not the ultimate goal, improving the Hessian convexity is. As a consequence, we are relaxed in finding precise samples, but place the emphasis on the Hessian eigenvalues.

Remark 3. *The optimal distribution is* $\tilde{\pi}(u) \propto \|G_\dagger(u)\|_2^2$*, but Theorem 2 does allow us to be different from it. In certain situations, the underlying inverse problem structure and some prior knowledge of* F *could potentially give some insights. For instance, in certain cases, one can show* $G_\dagger(u)$ *is uniformly bounded above and below for all* u*. When this happens, choosing a uniform distribution for* π_0 *may already give a satisfying sampling result. This is confirmed in our numerical test, seen in Figure 6.*

4. Application to the Schrödinger potential reconstruction

In this section, we demonstrate the performance of the proposed algorithm on a specific example: inverse steady state Schrödinger equation. The spacial domain is set to be $X = [-1; 1]^2 \subset \mathbb{R}^2$ and the time-independent PDE with constant source term writes as:

$$\begin{aligned} (-\Delta + p)u_p &= 10^5 \quad x \in X; \\ u_p &= 0 \quad x \in \partial X: \end{aligned} \tag{19}$$

The source term is set to be large only for readability of subsequent computation. The inverse problem is to reconstruct the potential p from measurements of the observable solution u_p. Clearly inferring p when the full knowledge of u_p is known is trivial: $p = \frac{1+\ u_p}{u_p}$ pointwise in X. The problem arises when p is assumed to be finite-dimensional (represented by finite many parameters) and only a finite number of measurements of u_p is taken. The goal is to find the optimal experimental setting to conduct such reconstruction.

Parameter Discretization. Let $f\ _k : X\ \ !\ \ \mathrm{R}g_{k=1;:::;K}$ be a given finite set of basis functions on X, and our admissible set for p is assume to be:

$$A := \quad p : X\ \ !\ \ \mathrm{R};\quad x = \begin{matrix} x_1 \\ x_2 \end{matrix}\ \ 7!\ \ p(x) = \sum_{k=1}^{K} p_k\ \ _k(x_1; x_2)$$

$$\text{for some } p_k\ 2\ \mathrm{R}; k = 1; :::; K\quad :$$

In the numerical examples in Section 4.1, we used $K = 9$ with corresponding basis

$$f\ \ _{k_1;k_2}(x_1; x_2) = \cos(k_1\ \ x_1)\cos(k_2\ \ x_2)g_{k_1;k_2=0;1;2}:$$

Experimental Setup. Without loss of generality, we assume all possible measurements are point-wise measurements, meaning $F(x; p) = u_p(x)$ for all $x\ 2\ X$. We denote the ground truth data generated by the ground truth media $fy(x) = F(x; p_?) = u_{p_?}(x)g$ with $p_?\ 2\ A$, so Assumption (A1) is satisfied. The question related to optimal design now translates to a search for the number c and locations $\Omega_c\quad X$ so to make the associated down-sampled optimization problem locally strictly convex.

Space discretization. To numerically realize the PDE solution, we use its numerical solution computed on equidistant Cartesian grid $f\ _n; n = 1; :::; (N_x + 1)^2g$, where we set N_x cells in every direction.

Numerical full measurement setup. The full measurement setup considers measurements taken at all vertices, meaning $\Omega = f\ _ng_{n=1}^{(N_x+1)^2}n@X$. So $j\Omega j = N = (N_x\quad 1)^2$. The cost function reads

$$C(p) = \frac{1}{2N}\sum_{x2} ju_p(x)\quad y(x)j^2\ :$$

Computation of $G_?(x)$. Evaluation of ˜ requires computation of the gradient $G_?(x)$ for all $x \in \Omega$, and in this context, it is $r_p u_{p?}(x)$ evaluated at $x \in \Omega$. In Appendix A we spell out the details of deploying an adjoint based method to compute the gradient. For example, the k-th entry of the gradient reveals

$$[G_?(x)]_k = [r_p u_p(x)]_k = \langle g^{(x)}; {}_k u_p \rangle_{L^2(X)};$$

where $g^{(x)}$ satisfies the adjoint equation

$$\Delta g^{(x)} + p g^{(x)} = {}_x \text{ on } X; \quad g^{(x)} = 0 \text{ on } \partial X: \tag{20}$$

Computationally both the forward and adjoint solvers are conducted by a finite element approach with nodal basis defined on an equidistant Cartesian grid $\{ {}_n \}$.

4.1. Importance Sampling Distributions

As a numerical study, we first run the equation with fine discretization, and plot out the optimal sampling strategy ˜. In the four examples shown, the $K = 9$ ground truth parameters are set according to Table 1. As shown in Figure 1, the optimal sampling distribution ˜ shows significant dependence on the underlying ground truth parameter.

System	ground truth parameter
A	$p_?^A = \begin{pmatrix} 13.6 & 10 & 10 \\ 10 & 10 & 10 \\ 10 & 10 & 10 \end{pmatrix}$
B	$p_?^B = \begin{pmatrix} 5.856 & 0.103 & 3.168 \\ 3.7441 & 2.493 & 1.124 \\ 0.9902 & 3.803 & 0.846 \end{pmatrix}$
C	$p_?^C = \begin{pmatrix} 11 & 8.889 & 7.778 \\ 6.667 & 5.556 & 4.444 \\ 3.333 & 2.222 & 1.111 \end{pmatrix}$
D	$p_?^D = \begin{pmatrix} 10 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Table 1: Test scenarios to study the optimal sampling strategy ˜. The $(i; j)$ entry of the matrix is the coe cient for p_k with $(k_1 = i; k_2 = j)$.

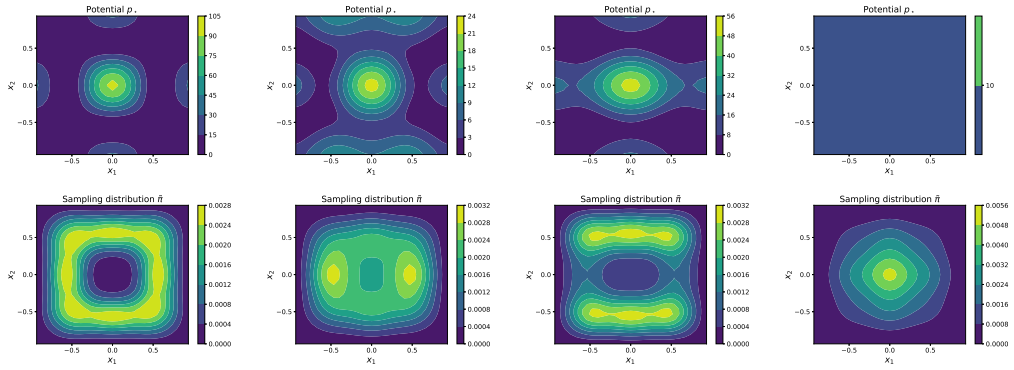


Figure 1: Top row shows four di erent ground-truth media $p_?$, and the bottom row shows the optimal sampling distribution ~ for each of them.

We then scale the parameters by multiplying $p_?$ with a scaling parameter . Varying the amplitude of , we observe very different pattern for ˜ as well, as shown in Figure 2. In this plot, we scale the ground truth distribution by constant (= 10 or 0:1) and we observe very different optimal distribution. Drawn from this numerical observation, we expect ˜ to be more centered in the middle when $p_?$ takes on small values, but develop interesting patterns when $p_?$ has a large scaling.

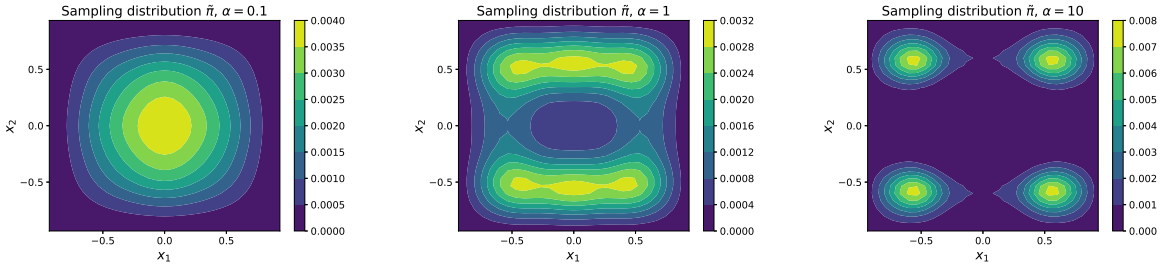
4.2. E ect of Sampling

As a proof of concept, we now study the performance of EKS sampling strategy for its recovery of optimal sensor locations. The minimal eigenvalue of the Hessian is a key quantity to be examined.

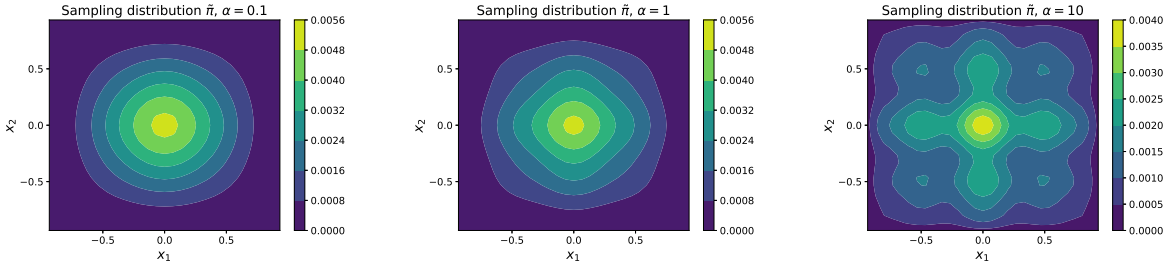
E ect on sensor locations and minimal Hessian eigenvalue. We choose the ground truth parameter of System C in Table 1 and use an adapted greedy version of EKS in [29] as described in Section 3.3 and a similar adaptation of CBS in [30].

To start, we evaluate the Hessian given by the full dataset. In Figure 3, with $N_x = 30$, we mark $N = (N_x \quad 1)^2 = 841$ red dots as the sensor locations and computed the optimal distribution ˜. The minimum eigenvalue of the Hessian in this setting is 70:21 > 0, and the problem is locally strictly convex.

To proceed with down-sampling, we allow only $c = 18 = 2K$ sensor locations. The initial guess was a normal distribution over Ω and the output is severely worse, with the minimal eigenvalue degenerated to 1:13e 2. Both



(a) System C.



(b) System D.

Figure 2: Optimal importance sampling distributions ~ for scaling parameters $p_?$ with = 0:1 (left), = 1 (center) and = 10 (right). The ground truth parameters $p_?$ from System C and D from Table 1 are taken.

EKS and CBS with greedy selection, after a running of 25 iterations, move the samples to new locations, and increase the minimum eigenvalue of the Hessian to 86:33 and 142:39, respectively. The samples drawn from the initial distribution, the iterated solution according to EKS and CBS are all plotted in Figure 4, and the evolution of the smallest eigenvalues and the Hessian error along iteration are plotted in Figure 5.

We note that the minimum eigenvalue for the Hessian generated by EKS and CBS are even larger than the minimum eigenvalue given by the full dataset. This suggests a vast number of data actually dilutes the information, making the landscape less convex than it can be by only using a smaller number of data points.

An interesting numerical discovery is that in this case, the uniformly distributed sensor locations, as depicted in Figure 6, also perform well, attaining a minimum eigenvalue of 102:03. Indeed the optimal importance sampling distribution ~ is bounded from above by 0:0031 (in comparison to $\frac{1}{N} =$ 0:0012) for a uniform distribution. Hence, the uniform distribution in

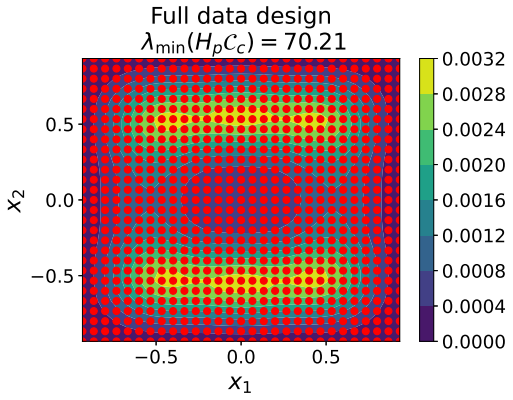


Figure 3: Full Data Setup: Measurement locations (red dots) are located in all grid points. The optimal importance sampling distribution ~ is drawn in the background.

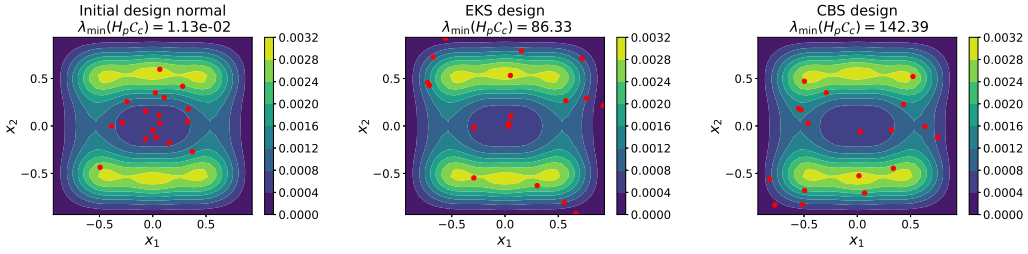


Figure 4: Red markers demonstrate the location of the sensors, with the background plotted as the optimal distribution. The left panel shows the distribution of the initial samples. The middle and the right panel show, respectively, the EKS and CBS samples after 25 iterations. The minimum eigenvalues of the Hessian change from 1e 2 to 86 and 142 respectively, ensuring local strong convexity.

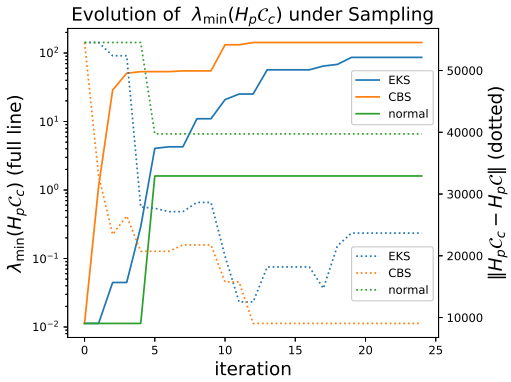


Figure 5: Evolution of minimum eigenvalue (solid lines) and deviation of the down-sampled Hessians from the full Hessian in Frobenius norm (dashed lines). Three sampling methods are used: EKS (blue), CBS (orange) and repeated sampling from the initial guess distribution (green), all used in greedy mode. Initial distribution is shared across three sampling methods.

this particular case is a good approximation (with 0:383). Starting from uniform distribution, we once again apply greedy EKS, CBS and repeated random sampling for 25 iterations and can further improve the eigenvalue to 204:88, 161:91 and 238:69, respectively.

E ect on the loss function. For a concrete visualization of the convexity improvement of the loss function, we confine ourselves to a two-dimensional admissible set with $A = fp : X \;!\; \mathsf{R}\, j\, p(x) = p_1 \cos(x_1) + p_2 \cos(x_2) g$ and the ground truth parameter $p_?(x_1; x_2) = 1 \cos(x_1) + 10 \cos(x_2)$. The profile of $p_?$ and the optimal importance sampling distribution are depicted in Figure 7. The scaling for $p_?$ in the x_1 and x_2 direction is very different, with $p_?$ changes its profile in x_2 direction significantly more. This is in alignment with the extension of the sampling probability.

When the full dataset if used, the loss function is convex, with the minimum eigenvalue being 4791:3, as shown in Figure 8. An initial setup of 8 normally distributed sensor locations shows significantly reduced convexity in the loss landscape, and the minimum eigenvalue becomes 301:85. Sampling with a greedy strategy in Figure 10 according to EKS and CBS enhances convexity dramatically, as plotted in Figure 10.

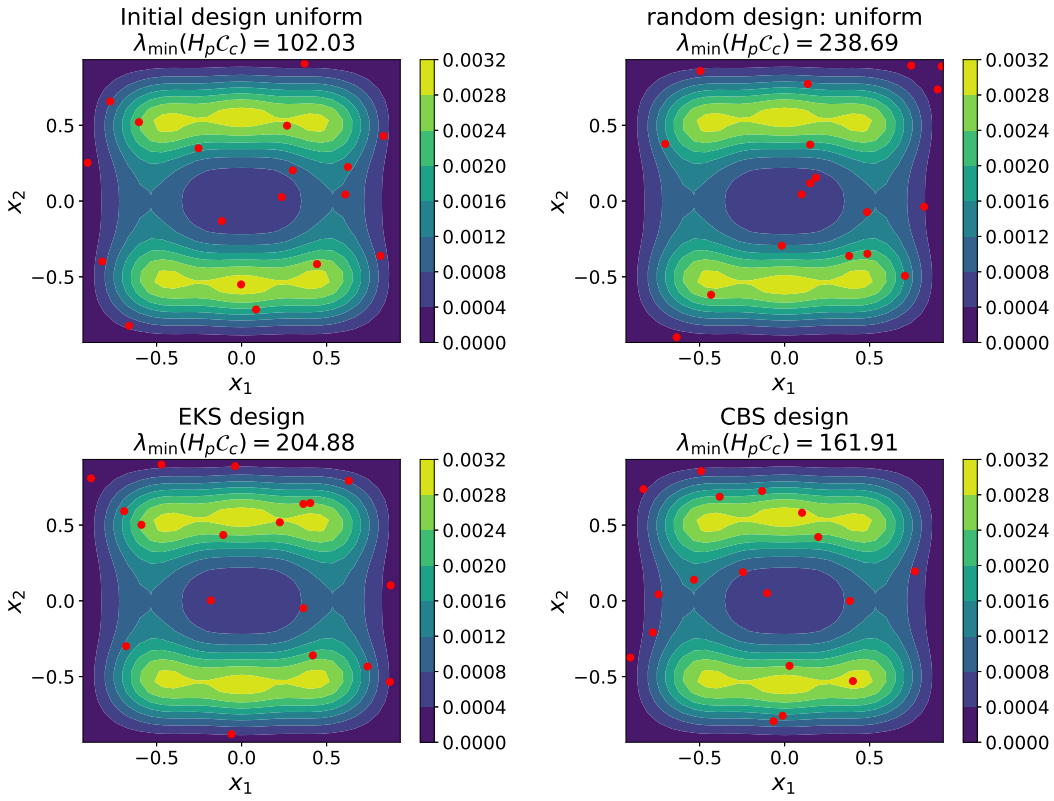


Figure 6: Uniformly distributed initial guess (upper left) of the distribution of the sensors (red dots) in the domain X, where the optimal importance sampling distribution is drawn in the background. Application of greedy EKS (lower left), CBS (lower left) and repeated sampling with respect to (w.r.t.) the uniform distribution changes the sensor distribution and dramatically increases the minimum eigenvalue of the Hessian of the cost function HC_c.

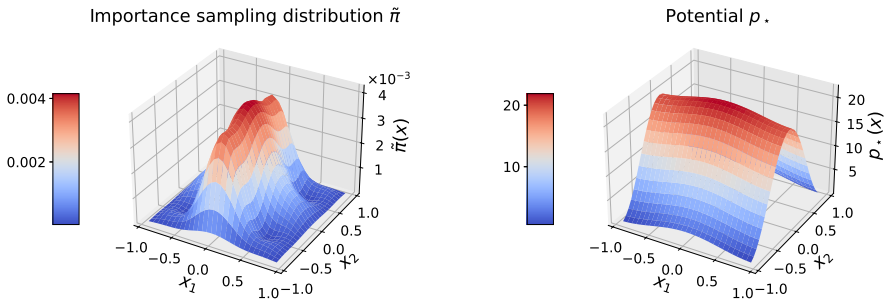


Figure 7: Optimal importance sampling distribution ~ (left) and shape of the ground truth parameter $p_?$ (right) in the two-dimensional setting.

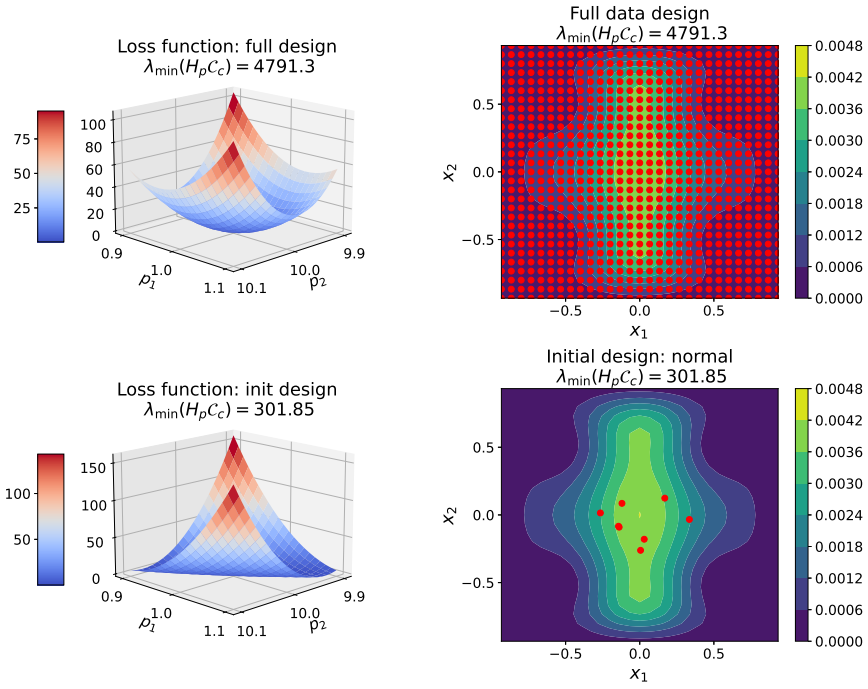


Figure 8: Loss landscapes (left) for di erent sensor locations (right): full data setup (rst row) and normally distributed initial sensor locations (second row).

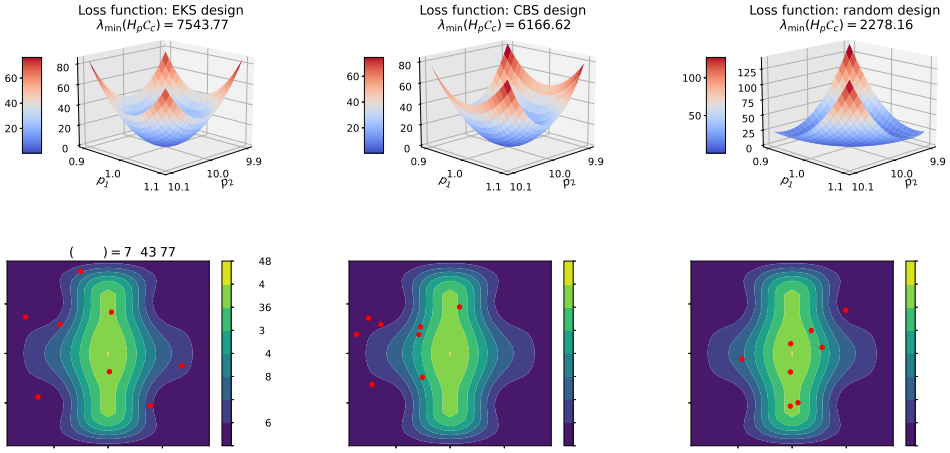


Figure 9: The top row shows the Loss landscapes and the bottom row shows the locations of the samples with the background presenting the optimal distribution. The three panels are results from greedy EKS, greedy CBS and repeated normal sampling from initial guess distribution.

5. Discussion

In this work, we study the unique reconstructability of a parameterized inverse problem with the perspective of preserving the positivity of the Gauss-Newton Hessian. Set in the framework of optimization problem, we translate the problem of examining the numerical reconstructability to that of convexity of the cost function at the global optimum point. Suppose the full data set provides the strict positivity of the Hessian, we examine how much one can down-sample the data. This problem is formulated as a matrix sketching problem, where a well-studied sketching algorithm from RNLA becomes handy. To down sample, the sample size depends on a sampling distribution that reflects the structure of the forward problem. To draw samples from this distribution, sampling algorithms such as EKS and CBS are implemented.

The general program described in this article can be applied to a variety of experimental design / data selection tasks merged from inverse problems. As a proof of concept, we provide a numerical test using Schrödinger equation as the forward model. The optimal distribution is problem dependent and is typically unavailable. In various applications, knowledge of the forward model can be used to obtain some qualitative estimates.

Following this work, many new questions can be asked. The paper

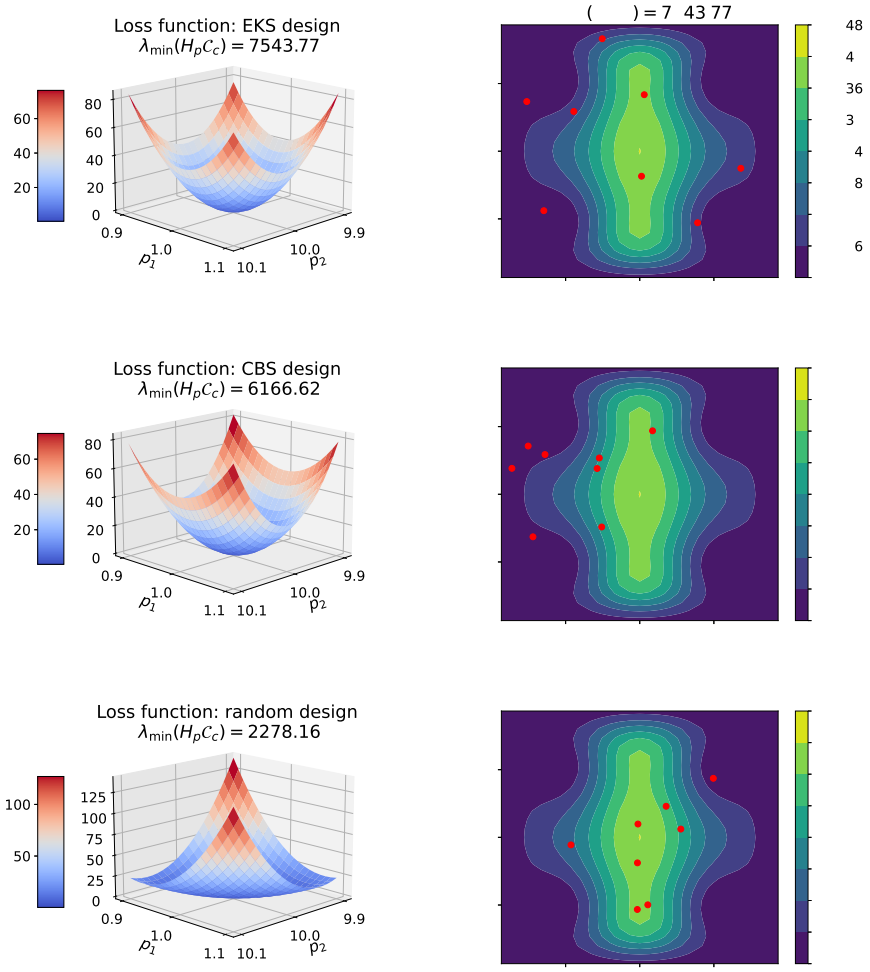


Figure 10: Loss landscapes (left) for di erent sensor locations (right) obtained by sampling according to greedy EKS (rst row), CBS (second row) and repeated normal sampling (last row) with initial setup in the second row of Figure 8.

presents result assuming the design space is finite with $j\Omega j = N < 1$, but Theorem 1 should be extendable to handle situations where the design space is infinite in size. In Bayesian optimal experimental design [4], E-optimality seeks to maximize the minimal eigenvalue of inverse of the covariance matrix, and thus evaluates the same quantity that we are examining in this work. Finding the explicit relation between the two approaches is also one of interesting future direction. Other optimal design criterion can also be considered. For example, one criterion to characterize convexity of the loss function is the K-optimality criterion [38], which minimizes the condition number of the covariance matrix. Running the greedy algorithm with this criterion could prevent flat looking loss functions due to different sizes of the eigenvalues of $H_p C$, as in the case of the last row of Figure 10. Furthermore, our presentation focuses on noise-free case. With noise present in data, by controlling the size of the noise, one can still argue the positivity of the Hessian around the global basin. The derivation is more convoluted and will be in our future research.

Finally, we see potential application of our approach to more recently developed inversion frameworks that rely on a least squares optimization. Examples of such frameworks can be found in [39, 40], where Gaussian processes or neural networks are incorporated in the inversion process. A detailed derivation is left for further investigation.

Appendix A. Appendix: Derivation of the formula for $\nabla_p u_p(x)$

We derive the formula for the gradient $r_p u_p(x)$ of the solution to the Schrödinger equation w.r.t. the potential p, that we require for the computation of the sampling probabilities.

In the following derivations, all gradients are with respect to x, unless specified otherwise. For a fixed measurement location $x \in X$, we can then define the Lagrange function $L : A \quad H_0^1(X) \quad H_0^1(X) \to R$ as

$$L_x(p; u; g) = u(x) + h r g; r u i_{L_2(X)} + hg; pu i_{L^2(X)} \quad hg; f i_{H^1(X);H^{-1}(X)};$$

where g is the Lagrange multiplier, and $h\,;\, i_{H_0^1(X);H^{-1}(X)}$ denotes the duality bracket in $H_0^1(X) \quad H^{-1}(X)$. Using (19), one immediately sees $L_x(p; u_p; g) = u_p(x)$. Therefore, confined on this solution manifold, chain rule gives:

$$\frac{@u_p(x)}{@p_j}_{p=\hat{p}} = \frac{@L_x}{@p_j}_{\substack{p=\hat{p} \\ u=u_{\hat{p}}}} + \frac{@L_x}{@u}_{\substack{p=\hat{p} \\ u=u_{\hat{p}}}} \frac{@u_p}{@p_j}_{p=\hat{p}} :$$

This equation holds valid for arbitrary g, and thus we would like to choose $g = g_x$ such that $@L_x=@u = 0$. If so:

$$\frac{@u_p(x)}{@p_j}\Big|_{p=\hat{p}} = \frac{@L_x}{@p_j}\Big|_{\substack{p=\hat{p}\\ u=\bar{P}u_{\hat{p}}}} = \frac{@hg_x; pui_{L^2(X)}}{@p_j}\Big|_{\substack{p=\hat{p}\\ u=u_{\hat{p}}}}$$
$$= \frac{@hg_x; \quad_k p_k \ _k ui_{L^2(X)}}{@p_j}\Big|_{\substack{p=\hat{p}\\ u=u_{\hat{p}}}} = hg_x; \quad_j u_{\hat{p}} i_{L^2(X)} :$$

It remains to compute $g_x \ 2 \ H_0^1(X)$ for which $@L_x(p; u; g_x)=@u = 0$. From integration by parts we see

$$@_u L_x = @_u \ u(x) + h \, r \, g_x; \ r \, ui_{L^2(X)} + hg_x; pui_{L^2(X)}$$
$$= @_u \ u(x) + h \quad \Delta g_x; ui_{H^{\ 1}(X);H^1(X)} + hpg_x; ui_{L^2(X)} \ :$$

Setting this to be zero, we have the condition for g_x:

$$\Delta g_x + pg_x = \quad_x \text{ on } X; \quad g_x = 0 \text{ on } @X :$$

Funding

K.H. acknowledges support by the German Academic Scholarship Foundation (Studienstiftung des deutschen Volkes) as well as the Marianne-Plehn-Program. C.K. acknowledges support from the German Science Foundation, KL566/22-1. Q.L. acknowledges support from DMS-2308440 and DMS-2023239.

References

[1] J. Chung, M. Chung, J. T. Slagel, Iterative sampled methods for massive and separable nonlinear inverse problems, in: Scale Space and Variational Methods in Computer Vision: 7th International Conference, SSVM 2019, Hofgeismar, Germany, June 30–July 4, 2019, Proceedings 7, Springer, 2019, pp. 119–130.

[2] J. Kiefer, Optimum experimental designs, Journal of the Royal Statistical Society. Series B (Methodological) 21 (2) (1959) 272–319.
URL http://www.jstor.org/stable/2983802

[3] T. J. Mitchell, An algorithm for the construction of "d-optimal" experimental designs, Technometrics 42 (1) (2000) 48–54.

[4] A. Alexanderian, Optimal experimental design for infinite-dimensional bayesian inverse problems governed by pdes: A review, Inverse Problems 37 (01 2021). doi:10.1088/1361-6420/abe10c.

[5] S. Bandara, J. P. Schlöder, R. Eils, H. G. Bock, T. Meyer, Optimal experimental design for parameter estimation of a cell signaling model, PLoS computational biology 5 (11) (2009) e1000558.

[6] S. Park, D. Kato, Z. Gima, R. Klein, S. Moura, Optimal experimental design for parameterization of an electrochemical lithium-ion battery model, Journal of The Electrochemical Society 165 (7) (2018) A1309.

[7] A. Attia, E. Constantinescu, Optimal experimental design for inverse problems in the presence of observation correlations, SIAM Journal on Scientific Computing 44 (4) (2022) A2808–A2842.

[8] X. Huan, J. Jagalur, Y. Marzouk, Optimal experimental design: Formulations and computations (2024). arXiv:2407.16212.
URL https://arxiv.org/abs/2407.16212

[9] M. W. Mahoney, Lecture notes on randomized linear algebra (2016). arXiv:1608.04481.

[10] P. Chen, Hessian matrix vs. gauss–newton hessian matrix, SIAM Journal on Numerical Analysis 49 (4) (2011) 1417–1435. doi:10.1137/100799988.

[11] C. Chen, S. Reiz, C. D. Yu, H.-J. Bungartz, G. Biros, Fast approximation of the gauss–newton hessian matrix for the multilayer perceptron, SIAM Journal on Matrix Analysis and Applications 42 (1) (2021) 165–184.

[12] D. S. Gonçalves, S. A. Santos, Local analysis of a spectral correction for the gauss-newton model applied to quadratic residual problems, Numerical algorithms 73 (2016) 407–431.

[13] J. Yang, J. Huang, Z. Li, H. Zhu, G. A. McMechan, X. Luo, Approximating the gauss–newton hessian using a space-wavenumber filter and

its applications in least-squares seismic imaging, IEEE Transactions on Geoscience and Remote Sensing 60 (2021) 1–13.

[14] T. Bui-Thanh, Q. Li, L. Zepeda-Núñez, Bridging and improving theoretical and computational electrical impedance tomography via data completion, SIAM Journal on Scientific Computing 44 (3) (2022) B668–B693. doi:10.1137/21M141703X.

[15] K. Manohar, B. W. Brunton, J. N. Kutz, S. L. Brunton, Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns, IEEE Control Systems Magazine 38 (3) (2018) 63–86. doi:10.1109/MCS.2018.2810460.

[16] R. Orozco, F. J. Herrmann, P. Chen, Probabilistic bayesian optimal experimental design using conditional normalizing flows, arXiv preprint arXiv:2402.18337 (2024).

[17] P.-G. Martinsson, J. A. Tropp, Randomized numerical linear algebra: Foundations and algorithms, Acta Numerica 29 (2020) 403–572. doi:10.1017/S0962492920000021.

[18] D. P. Woodruff, et al., Sketching as a tool for numerical linear algebra, Foundations and Trends® in Theoretical Computer Science 10 (1–2) (2014) 1–157.

[19] Z. Huang, S. Becker, Spectral estimation from simulations via sketching, Journal of Computational Physics 447 (2021) 110686.

[20] A. S. Dalalyan, A. Karagulyan, User-friendly guarantees for the langevin monte carlo with inaccurate gradient, Stochastic Processes and their Applications 129 (12) (2019) 5278–5311.

[21] G. O. Roberts, R. L. Tweedie, Exponential convergence of langevin distributions and their discrete approximations, Bernoulli 2 (4) (1996) 341–363.
URL http://www.jstor.org/stable/3318418

[22] X. Cheng, N. S. Chatterji, P. L. Bartlett, M. I. Jordan, Underdamped langevin mcmc: A non-asymptotic analysis, in: Conference on learning theory, PMLR, 2018, pp. 300–323.

[23] T. Chen, E. Fox, C. Guestrin, Stochastic gradient hamiltonian monte carlo, in: International conference on machine learning, PMLR, 2014, pp. 1683–1691.

[24] O. Mangoubi, A. Smith, Mixing of hamiltonian monte carlo on strongly log-concave distributions: Continuous dynamics, The Annals of Applied Probability 31 (5) (2021) 2019–2045.

[25] N. Bou-Rabee, M. Hairer, Nonasymptotic mixing of the mala algorithm, IMA Journal of Numerical Analysis 33 (1) (2013) 80–110.

[26] R. Dwivedi, Y. Chen, M. J. Wainwright, B. Yu, Log-concave sampling: Metropolis-hastings algorithms are fast, Journal of Machine Learning Research 20 (183) (2019) 1–42.

[27] S. Reich, A dynamical systems framework for intermittent data assimilation, BIT Numerical Mathematics 51 (2011) 235–249.

[28] G. Evensen, F. C. Vossepoel, P. J. Van Leeuwen, Data assimilation fundamentals: A unified formulation of the state and parameter estimation problem, Springer Nature, 2022.

[29] A. Garbuno-Inigo, F. Hoffmann, W. Li, A. M. Stuart, Interacting langevin diffusions: Gradient structure and ensemble kalman sampler, SIAM Journal on Applied Dynamical Systems 19 (1) (2020) 412–441. doi:10.1137/19M1251655.

[30] J. A. Carrillo, F. Hoffmann, A. M. Stuart, U. Vaes, Consensus-based sampling, Studies in Applied Mathematics 148 (3) (2022) 1069–1140. doi:https://doi.org/10.1111/sapm.12470.

[31] Z. Ding, Q. Li, Ensemble kalman sampler: Mean-field limit and convergence analysis, SIAM Journal on Mathematical Analysis 53 (2) (2021) 1546–1578. doi:10.1137/20M1339507.

[32] U. Vaes, Sharp propagation of chaos for the ensemble langevin sampler (2024). arXiv:2404.06456.

[33] Z. Shun, P. McCullagh, Laplace approximation of high dimensional integrals, Journal of the Royal Statistical Society Series B: Statistical Methodology 57 (4) (1995) 749–760.

[34] K. Riedl, T. Klock, C. Geldhauser, M. Fornasier, Gradient is all you need? (2023). arXiv:2306.09778.
URL https://arxiv.org/abs/2306.09778

[35] B. Sprungk, S. Weissmann, J. Zech, Metropolis-adjusted interacting particle sampling (2023). arXiv:2312.13889.
URL https://arxiv.org/abs/2312.13889

[36] K. Hellmuth, C. Klingenberg, Q. Li, M. Tang, Numerical reconstruction of the kinetic chemotaxis kernel from macroscopic measurement, wellposedness and illposedness (2023). arXiv:2309.05004.

[37] N. B. Kovachki, A. M. Stuart, Ensemble kalman inversion: a derivative-free technique for machine learning tasks, Inverse Problems 35 (9) (2019) 095005. doi:10.1088/1361-6420/ab1c3a.

[38] J. J. Ye, J. Zhou, Minimizing the condition number to construct design points for polynomial regression models, SIAM Journal on Optimization 23 (1) (2013) 666–686.

[39] Y. Chen, B. Hosseini, H. Owhadi, A. M. Stuart, Solving and learning nonlinear pdes with gaussian processes, Journal of Computational Physics 447 (2021) 110668.

[40] S. Dong, Y. Wang, A method for computing inverse parametric pde problems with random-weight neural networks, Journal of Computational Physics 489 (2023) 112263.