SNIPS: Solving Noisy Inverse Problems Stochastically

Bahjat Kawar, Gregory Vaksman, Michael Elad
Computer Science Department, Technion, Haifa, Israel
{bahjat.kawar, grishav, elad}@cs.technion.ac.il

Abstract

In this work we introduce a novel stochastic algorithm dubbed SNIPS, which draws samples from the posterior distribution of any linear inverse problem, where the observation is assumed to be contaminated by additive white Gaussian noise. Our solution incorporates ideas from Langevin dynamics and Newton’s method, and exploits a pre-trained minimum mean squared error (MMSE) Gaussian denoiser. The proposed approach relies on an intricate derivation of the posterior score function that includes a singular value decomposition (SVD) of the degradation operator, in order to obtain a tractable iterative algorithm for the desired sampling. Due to its stochasticity, the algorithm can produce multiple high perceptual quality samples for the same noisy observation. We demonstrate the abilities of the proposed paradigm for image deblurring, super-resolution, and compressive sensing. We show that the samples produced are sharp, detailed and consistent with the given measurements, and their diversity exposes the inherent uncertainty in the inverse problem being solved.

1 Introduction

Many problems in the field of image processing can be cast as noisy linear inverse problems. This family of tasks includes denoising, inpainting, deblurring, super resolution, compressive sensing, and many other image recovery problems. A general linear inverse problem is posed as

\[ y = Hx + z, \]

where we aim to recover a signal \( x \) from its measurement \( y \), given through a linear degradation operator \( H \) and a contaminating noise, being additive, white and Gaussian, \( z \sim N(0, \sigma_0^2 I) \). In this work we assume that both \( H \) and \( \sigma_0 \) are known.

Over the years, many strategies, algorithms and underlying statistical models were developed for handling image restoration problems. A key ingredient in many of the classic attempts is the prior that aims to regularize the inversion process and lead to visually pleasing results. Among the various options explored, we mention sparsity-inspired techniques [13, 55, 11], local Gaussian-mixture modeling [57, 63], and methods relying on non-local self-similarity [6, 9, 36, 51]. More recently, and with the emergence of deep learning techniques, a direct design of the recovery path from \( y \) to an estimate of \( x \) took the lead, yielding state-of-the-art results in various linear inverse problems, such as denoising [23, 59, 61, 52], deblurring [22, 48], super resolution [10, 17, 54] and other tasks [29, 28, 19, 16, 37, 58].

Despite the evident success of the above techniques, many image restoration algorithms still have a critical shortcoming: In cases of severe degradation, most recovery algorithms tend to produce washed out reconstructions that lack details. Indeed, most image restoration techniques seek a reconstruction that minimizes the mean squared error between the restored image, \( \hat{x} \), and the unknown original one, \( x \). When the degradation is acute and information is irreversibly lost, image reconstruction becomes a highly ill-posed problem, implying that many possible clean images could explain the given
We refer hereafter to our algorithm as SNIPS when perceptual quality becomes our prime objective, the strategy for solving inverse problems must necessarily change. More specifically, the solution should concentrate on producing a sample (or many of them) from the posterior distribution \( p(\mathbf{x}|\mathbf{y}) \) instead of its conditional mean. Recently, two such approaches have been suggested – GAN-based and Langevin sampling. Generative Adversarial Networks (GANs) have shown impressive results in generating realistically looking images (e.g., \[14,33\]). GANs can be utilized for solving inverse problems while producing high-quality images (see e.g., \[2,31,44\]). These solvers aim to produce a diverse set of output images that are consistent with the measurements, while also being aligned with the distribution of clean examples. A major disadvantage of GAN-based algorithms for inverse problems is their tendency (as practiced in \[2,31,44\]) to assume noiseless measurements, a condition seldom met in practice. An exception to this is the work reported in \[33\], which adapts a conditional GAN to become a stochastic denoiser.

The second approach for sampling from the posterior, and the one we shall be focusing on in this paper, is based on Langevin dynamics. This core iterative technique enables sampling from a given distribution by leveraging the availability of the score function – the gradient of the log of the probability density function \([38,3]\). The work reported in \[44,20,46\] utilizes the annealed Langevin dynamics method, both for image synthesis and for solving noiseless inverse problems.

Their synthesis algorithm relies on an MMSE Gaussian denoiser (given as a neural network) for approximating a gradually blurred score function. In their treatment of inverse problems, the conditional score remains tractable and manageable due to the noiseless measurements assumption.

The question addressed in this paper is the following: How can the above line of Langevin-based work be generalized for handling linear inverse problems, as in \[\text{Equation 1}\], in which the measurements are noisy? A partial and limited answer to this question has already been given in \[21\] for the tasks of image denoising and inpainting. The present work generalizes these \([44,20,46,21]\) results, and introduces a systematic way for sampling from the posterior distribution of any given noisy linear inverse problem. As we carefully show, this extension is far from being trivial, due to two prime reasons: (i) The involvement of the degradation operator \( \mathbf{H} \), which poses a difficulty for establishing a relationship between the reconstructed image and the noisy observation; and (ii) The intricate connection between the measurements and the synthetic annealed Langevin noise. Our proposed remedy is to decorrelate the measurements equation via a singular value decomposition (SVD) of the operator \( \mathbf{H} \), which decouples the dependencies between the measurements, enabling each to be addressed by an adapted iterative process. In addition, we define the annealing noise to be built as portions of the measurement noise itself, in a manner that facilitates a constructive derivation of the conditional score function.

Following earlier work \([44,20,46,21]\), our algorithm is initialized with a random noise image, gradually converging to the reconstructed result, while following the direction of the log-posterior gradient, estimated using an MMSE denoiser. Via a careful construction of the gradual annealing noise sequence, from very high values to low ones, the entries in the derived score switch mode. Those referring to non-zero singular values start by being purely dependent on the measurements, and then transition to incorporate prior information based on the denoiser. As for entries referring to zero singular values, their corresponding entries undergo a pure synthesis process based on the prior-only score function. Note that the denoiser blends values in the evolving sample, thus intermixing the influence of the gradient entries. Our derivations include an analytical expression for a position-dependent step size vector, drawing inspiration from Newton’s method in optimization. This stabilizes the algorithm and is shown to be essential for its success.

We refer hereafter to our algorithm as SNIPS (Solution of Noisy Inverse Problems Stochastically). Observe that as we target to sample from the posterior distribution \( p(\mathbf{x}|\mathbf{y}) \), different runs of SNIPS on the same input necessarily yield different results, all of which valid solutions to the given inverse problem.

\[1\] The work reported in \([18,42,28]\) and \([15,23]\) is also relevant to this discussion, but somewhat different. We shall specifically address these papers’ content and its relation to our work in \text{section 2}\.
We demonstrate SNIPS on image deblurring, single image super resolution, and compressive sensing, all of which contain non-negligible noise, and emphasize the high perceptual quality of the results, their diversity, and their relation to the MMSE estimate.

To summarize, this paper’s contributions are threefold:

- We present an intricate derivation of the blurred posterior score function for general noisy inverse problems, where both the measurement and the target image contain delicately inter-connected additive white Gaussian noise.
- We introduce a novel stochastic algorithm – SNIPS – that can sample from the posterior distribution of these problems. The algorithm relies on the availability of an MMSE denoiser.
- We demonstrate impressive results of SNIPS on image deblurring, single image super resolution, and compressive sensing, all of which are highly noisy and ill-posed.

Before diving into the details of this work, we should mention that using Gaussian denoisers iteratively for handling general linear inverse problems has already been proposed in the context of the Plug-and-Play-Prior (PnP) method [53] and RED [39], and their many followup papers (e.g., [60], [30], [1], [49], [7], [50], [40], [4]). However, both PnP and RED are quite different from our work, as they do not target sampling from the posterior, but rather focus on MAP or MMSE estimation.

2 Background

The Langevin dynamics algorithm [3], [38] suggests sampling from a probability distribution \( p(x) \) using the iterative transition rule

\[
x_{t+1} = x_t + \alpha \nabla_x \log p(x_t) + \sqrt{2\alpha} z_t,
\]

where \( z_t \sim \mathcal{N}(0, I) \) and \( \alpha \) is an appropriately chosen small constant. The added \( z_t \) allows for stochastic sampling, avoiding a collapse to a maximum of the distribution. Initialized randomly, after a sufficiently large number of iterations, and under some mild conditions, this process converges to a sample from the desired distribution \( p(x) \) [38].

The work reported in [44] extends the aforementioned algorithm into annealed Langevin dynamics. The annealing proposed replaces the score function in Equation 2 with a blurred version of it, \( \nabla_{\tilde{x}} \log p(\tilde{x}_t) \), where \( \tilde{x}_t = x_t + n \) and \( n \sim \mathcal{N}(0, \sigma^2 I) \) is a synthetically injected noise. The core idea is to start with a very high noise level \( \sigma \) and gradually drop it to near-zero, all while using a step size \( \alpha \) dependent on the noise level. These changes allow the algorithm to converge much faster and perform better, because it widens the basin of attraction of the sampling process. The work in [20] further develops this line of work by leveraging a brilliant relation attributed to Miyasawa [32] (also known as Stein’s integration by parts trick [47] or Tweedie’s identity [12]). It is given as

\[
\nabla_{\tilde{x}_t} \log p(\tilde{x}_t) = \frac{D(\tilde{x}_t, \sigma) - \tilde{x}_t}{\sigma^2},
\]

where \( D(\tilde{x}_t, \sigma) \) is a function of the noise level. This allows for efficient sampling from the posterior distribution.
We consider the problem of recovering a signal $x$. When turning to solve inverse problems, previous work suggests sampling from the posterior distribution $p(x|y)$ using annealed Langevin dynamics \cite{20, 45, 21} or similar methods \cite{15, 18, 42, 26}, by replacing the score function used in the generation algorithm with a conditional one. As explained in the supplemental material, the sampling should be performed in the SVD domain $\sigma$ with noise levels $\tilde{x}$ where we have relied on the relation $\tilde{x} = Hx$ for specific and simplified choices of $H$ and with no noise in the measurements. The works in \cite{15, 25} avoid these difficulties altogether by returning to the original (non-annealed) Langevin method, with the unavoidable cost of becoming extremely slow. In addition, their algorithms are demonstrated on inverse problems in which the additive noise is restricted to be very weak. The work in \cite{21} is broader, allowing for an arbitrary additive white Gaussian noise, but limits $H$ to the problems of denoising or inpainting. While all these works demonstrate high quality results, there is currently no clear way for deriving the blurred score function of a general linear inverse problem as posed in Equation 1. In the following, we present such a derivation.

3 The Proposed Approach: Deriving the Conditional Score Function

3.1 Problem Setting

We consider the problem of recovering a signal $x \in \mathbb{R}^N$ (where $x \sim p(x)$ and $p(x)$ is unknown) from the observation $y = Hx + z$, where $y \in \mathbb{R}^M$, $H \in \mathbb{R}^{M \times N}$, $M \leq N$, $z \sim \mathcal{N}(0, \sigma^2 I)$, and $H$ and $\sigma_0$ are known. Our ultimate goal is to sample from the posterior $p(x|y)$. However, since access to the score function $\nabla_x \log p(x|y)$ is not available, we re-target our goal, as explained above, to sampling from blurred posterior distributions, $p(\tilde{x}|y)$, where $\tilde{x} = x + n$ and $n \sim \mathcal{N}(0, \sigma^2 I)$, with noise levels $\sigma$ starting very high, and decreasing towards near-zero.

As explained in the supplemental material, the sampling should be performed in the SVD domain in order to get a tractable derivation of the blurred score function. Thus, we consider the singular value decomposition (SVD) of $H$, given as $H = U \Sigma V^T$, where $U \in \mathbb{R}^{M \times M}$ and $V \in \mathbb{R}^{N \times N}$ are orthogonal matrices, and $\Sigma \in \mathbb{R}^{M \times N}$ is a rectangular diagonal matrix containing the singular values of $H$, denoted as $s_j \_{j=1}^M$ in descending order ($s_1 > s_2 > \cdots > s_{M-1} > s_M \geq 0$). For convenience of notations, we also define $s_j = 0$ for $j = M + 1, \ldots, N$. To that end, we notice that

$$p(\tilde{x}|y) = p(\tilde{x}|U^Ty) = p(V^T\tilde{x}|U^Ty).$$

The first equality holds because the multiplication of $y$ by the orthogonal matrix $U^T$ does not add or remove information, and the second equality holds because the multiplication of $\tilde{x}$ by $V^T$ does not change its probability \cite{24}. Therefore, sampling from $p(V^T\tilde{x}|U^Ty)$ and then multiplying the result by $V$ will produce the desired sample from $p(\tilde{x}|y)$. As we are using Langevin dynamics, we need to calculate the conditional score function $\nabla_{V^T\tilde{x}} \log p(V^T\tilde{x}|U^Ty)$. For simplicity, we denote hereafter $y_T = U^Ty$, $z_T = U^Tz$, $x_T = V^Tx$, $n_T = \Sigma V^Tn$, and $\tilde{x}_T = V^T\tilde{x}$. Observe that with these notations, the measurements equation becomes

$$y = Hx + z = \Sigma \Sigma V^T x + z,$$

and thus

$$U^Ty = \Sigma V^T x + U^Tz = \Sigma V^T(\tilde{x} - n) + U^Tz = \Sigma V^T \tilde{x} - \Sigma V^T n + U^Tz,$$

where we have relied on the relation $\tilde{x} = x + n$. This leads to

$$y_T = \Sigma \Sigma x_T - n_T + z_T.$$

In this formulation, which will aid in deriving the conditional score, our aim is to make design choices on $n_T$ such that $z_T - n_T$ has uncorrelated entries and is independent of $x_T$. This brings us to the formation of the synthetic annealed noise, which is an intricate ingredient in our derivations.

\footnote{We assume $M \leq N$ for ease of notations, and because this is the common case. However, the proposed approach and all our derivations work just as well for $M > N$.}
We base this formulation on the definition of a sequence of noise levels \( \{ \sigma_i \}_{i=1}^{L+1} \) such that \( \sigma_1 > \sigma_2 > \cdots > \sigma_L > \sigma_{L+1} = 0 \), where \( \sigma_1 \) is high (possibly \( \sigma_1 > |x|_\infty \)) and \( \sigma_L \) is close to zero. We require that for every \( j \) such that \( s_j \neq 0 \), there exists \( i_j \) such that \( \sigma_{i_j} s_j < \sigma_0 \) and \( \sigma_{i_j-1} s_j > \sigma_0 \). This implies \( \forall i : \sigma_i s_j \neq \sigma_0 \), which helps ease notations. SNIPS works just as well for \( \sigma_i s_j = \sigma_0 \).

Using \( \{ \sigma_i \}_{i=1}^{L+1} \), we would like to define \( \{ \tilde{x}_i \}_{i=1}^{L+1} \), a sequence of noisy versions of \( x \), where the noise level in \( \tilde{x}_i \) is \( \sigma_i \). One might be tempted to define these noise additions as independent of the measurement noise \( z \). However, this option leads to a conditional score term that cannot be calculated analytically, as explained in the supplemental material. Therefore, we define these noise additions differently, as carved from \( z \) in a gradual fashion. To that end, we define \( \tilde{x}_{L+1} = x \), and for every \( i = L, L-1, \ldots, 1 : \tilde{x}_i = \tilde{x}_{i+1} + \eta_i \), where \( \eta_i \sim N(0, (\sigma_i^2 - \sigma_{i+1}^2) I) \). This results in \( \tilde{x}_i = x + n_i \), where \( n_i = \sum_{k=i}^{L} \eta_k \sim N(0, \sigma_i^2 I) \).

And now we turn to define the statistical dependencies between the measurements’ noise \( z \) and the artificial noise vectors \( \eta_i \). Since \( \eta_i \) and \( z \) are each Gaussian with uncorrelated entries, so are the components of the vectors \( \Sigma V^T \eta_i \), \( \Sigma V^T n_i \), and \( z_T \). In order to proceed while easing notations, let us focus on a single entry \( j \) in these three vectors, for which \( s_j > 0 \), and omit this index. We denote these entries as \( \eta_{T,j}, n_{T,j} \) and \( z_T \), respectively. We construct \( \eta_{T,j} \) such that

\[
E[\eta_{T,j} \cdot z_T] = \begin{cases} 
E[\eta_{T,j}] & \text{for } i \geq i_j \\
(z_T - n_{T,j})^2 & \text{for } i = i_j - 1 \\
0 & \text{otherwise.}
\end{cases}
\]

This implies that the layers of noise \( \eta_{T,L+1}, \ldots, \eta_{T,i_j} \) are all portions of \( z_T \) itself, with an additional portion being contained in \( \eta_{T,i_j-1} \). Afterwards, \( \eta_{T,i} \) become independent of \( z_T \). In the case of \( s_j = 0 \), the above relations simplify to be \( E[\eta_{T,j} \cdot z_T] = 0 \) for all \( i \), implying no statistical dependency between the given and the synthetic noises. Consequently, it can be shown that the overall noise in Equation 5 satisfies

\[
(\Sigma V^T n_i - z_T)_j = n_{T,j} - z_T \sim \begin{cases} 
N(0, s_j^2 \sigma_i^2 - \sigma_0^2) & \text{if } \sigma_i s_j > \sigma_0 \\
N(0, \sigma_0^2 - s_j^2 \sigma_i^2) & \text{otherwise.}
\end{cases}
\]

The top option refers to high values of the annealed Langevin noise, in which, despite the possible decay caused by the singular value \( s_j \), this noise is stronger than \( z_T \). In this case, \( n_{T,j} \) contains all \( z_T \) and an additional independent portion of noise. The bottom part assumes that the annealed noise (with the influence of \( s_j \)) is weaker than the measurements’ noise, and then it is fully immersed within \( z_T \), with the difference being Gaussian and independent.

### 3.2 Derivation of the Conditional Score Function

The above derivations show that the noise in Equation 5 is zero-mean, Gaussian with uncorrelated entries and of known variance, and this noise is independent of \( \tilde{x}_i \). Thus Equation 5 can be used conveniently for deriving the measurements part of the conditional score function. We denote \( \tilde{x}_T = V^T \tilde{x}_i \), \( \tilde{x} = \tilde{x}_i \), \( n = n_i \) for simplicity, and turn to calculate \( \nabla \tilde{x}_T \log p(\tilde{x}_T|y_T) \). We split \( \tilde{x}_T \) into three parts: (i) \( \tilde{x}_{T,0} \) refers to the entries \( j \) for which \( s_j = 0 \); (ii) \( \tilde{x}_{T,<} \) corresponds to the entries \( j \) for which \( 0 < \sigma_i s_j < \sigma_0 \); and (iii) \( \tilde{x}_{T,>} \) includes the entries \( j \) for which \( \sigma_i s_j > \sigma_0 \). Observe that this partition of the entries of \( \tilde{x}_T \) is non-overlapping and fully covering. Similarly, we partition every vector \( v \in R^N \) into \( v_0, v_<, v_> \), which are the entries of \( v \) corresponding to \( \tilde{x}_{T,0}, \tilde{x}_{T,<}, \tilde{x}_{T,>} \), respectively. Furthermore, we define \( v_0, v_<, v_> \) as all the entries of \( v \) except \( v_0, v_<, v_> \), respectively. With these definitions in place, the complete derivation of the score function is detailed in the supplemental material, and here we bring the final outcome. For \( \tilde{x}_{T,0} \), the score is independent of the measurements and given by

\[
\nabla \tilde{x}_{T,0} \log p(\tilde{x}_T|y_T) = (V^T \nabla \tilde{x} \log p(\tilde{x}))(0).
\]

For the case of \( \tilde{x}_{T,>} \), the expression obtained is only measurements-dependent,

\[
\nabla \tilde{x}_{T,>} \log p(\tilde{x}_T|y_T) = \left( \Sigma^T \left( \sigma_i^2 \Sigma \Sigma^T - \sigma_0^2 I \right) \right) \left( y_T - \Sigma \tilde{x}_T \right)_>.
\]
was approximated using a Quasi-Newton method. In our case, we analytically calculate a diagonal
where the conditional score function is estimated as described in subsection 3.2, and
which considerably slows down the algorithm.

Armed with the conditional score function in Equation 10, the Langevin dynamics algorithm can be
\[ \nabla_{\tilde{x}_{T}}, \log p(\tilde{x}_{T}|y_{T}) = \left( \Sigma^{T} \left( \sigma_{0}^{2} I - \sigma_{t}^{2} \Sigma \Sigma^{T} \right) \right)^{\dagger} (y_{T} - \Sigma \tilde{x}_{T}) \right) \_ \_ + (\nabla^{T} \Sigma \log p(\tilde{x})) \_ \_. \]  (9)

As already mentioned, the full derivations of equations 7, 8, and 9 are detailed in the supplemental
material. Aggregating all these results together, we obtain the following conditional score function:
\[ \nabla_{\tilde{x}_{T}}, \log p(\tilde{x}_{T}|y_{T}) = \Sigma^{T} \left( \sigma_{0}^{2} I - \sigma_{t}^{2} \Sigma \Sigma^{T} \right)^{\dagger} (y_{T} - \Sigma \tilde{x}_{T}) + (\nabla^{T} \Sigma \log p(\tilde{x})) \_ \_. \]  (10)

where \((\nu)|_{\tilde{x}}\) is the vector \(\nu\), but with zeros in its entries that correspond to \(\nu_{>}\). Observe that the
first term in Equation 10 contains zeros in the entries corresponding to \(\tilde{x}_{T,0}\), matching the above
calculations. The vector \(\nabla_{\tilde{x}} \log p(\tilde{x})\) can be estimated using a neural network as in [44], or using a
pre-trained MMSE denoiser as in [20, 21]. All the other elements of this vector are given or can be
easily obtained from \(H\) by calculating its SVD decomposition once at the beginning.

4 The Proposed Algorithm

Armed with the conditional score function in \textbf{Equation 10} the Langevin dynamics algorithm can be
run with a constant step size or an annealed step size as in [44], and this should converge to a sample
from \(p(\tilde{x}_{T}|y_{T})\). However, for this to perform well, one should use a very small step size, implying a
devastatingly slow convergence behavior. This is mainly due to the fact that different entries of \(\tilde{x}_{T}\)
advance at different speeds, in accord with their corresponding singular values. As the added noise
in each step has the same variance in every entry, this leads to an unbalanced signal-to-noise ratio,
which considerably slows down the algorithm.

In order to mitigate this problem, we suggest using a \textbf{step size vector} \(\alpha_{i} \in \mathbb{R}^{N}\). We denote
\(A_{i} = \text{diag} (\alpha_{i})\), and obtain the following update formula for a Langevin dynamics algorithm:
\[ \nabla^{T} \tilde{x}_{i} = \nabla^{T} \tilde{x}_{i-1} + c \cdot A_{i} \cdot \nabla^{T} \tilde{x}_{i} \_ \_ \log p(\nabla^{T} \tilde{x}_{i}|y_{T}) + \sqrt{2 \cdot c A_{i} \frac{1}{2} \cdot z_{i}}, \]  (11)

where the conditional score function is estimated as described in \textbf{subsection 3.2}, and \(c\) is some
constant. For the choice of the step sizes in the diagonal of \(A_{i}\), we draw inspiration from Newton’s
method in optimization, which is designed to speed up convergence to local maximum points. The
update formula in Newton’s method is the same as \textbf{Equation 11} but without the additional noise
\(z_{i}\), and with \(A_{i}\) being the negative inverse Hessian of \(\log p(\nabla^{T} \tilde{x}_{i}|y_{T})\). We calculate a diagonal
approximation of the Hessian, and set \(A_{i}\) to be its negative inverse. We also estimate the conditional
score function using \textbf{Equation 10} and a neural network. Note that this mixture of Langevin dynamics
and Newton’s method has been suggested in a slightly different context in [43], where the Hessian
was approximated using a Quasi-Newton method. In our case, we analytically calculate a diagonal
approximation of the negative inverse Hessian and obtain the following:
\[ (\alpha_{i})_{j} = \begin{cases} \sigma_{0}^{2} + s_{j}, & s_{j} = 0 \\ \sigma_{0}^{2} - \frac{s_{j}}{\sigma_{0}^{2}}, & \sigma_{t} s_{j} > \sigma_{0} \\ \sigma_{t}^{2} \cdot \frac{1 - s_{j}}{\sigma_{0}^{2}}, & 0 < \sigma_{t} s_{j} < \sigma_{0}. \end{cases} \]  (12)
The full derivations for each of the three cases are detailed in the supplemental material. Using these step sizes, the update formula in Equation 11, the conditional score function in Equation 10 and a neural network $s(\tilde{x}, \sigma)$ that estimates the score function $\nabla_{\xi} \log p(\tilde{x})$ we obtain a tractable iterative algorithm for sampling from $p(\tilde{x}|y)$, where the noise in $\tilde{x}$ is sufficiently negligible to be considered as a sampling from the ideal image manifold.

Algorithm 1: SNIPS

Input: $\{\sigma_i\}_{i=1}^{L}$, c, $\tau$, y, H, $\sigma_0$

1. $U, \Sigma, V \leftarrow \text{svd}(H)$
2. Initialize $x_0$ with random noise $U[0, 1]$
3. for $i \leftarrow 1$ to $L$
4.   $$(A_i)_0 \leftarrow \sigma_i^2 \mathbf{I}$$
5.   $$(A_i)_< \leftarrow \sigma_i^2 \cdot \left( \mathbf{I} - \frac{\sigma_i^2}{\sigma_0^2} \Sigma_{<} \Sigma_{<}^T \right)$$
6.   $$(A_i)_> \leftarrow \sigma_i^2 \mathbf{I} - \sigma_0^2 \Sigma_{>} \Sigma_{>}^T$$
7. for $t \leftarrow 1$ to $\tau$
8.   Draw $z_t \sim \mathcal{N}(0, \mathbf{I})$
9.   $$d_t \leftarrow \Sigma d_{t-1} + \left(\frac{A_i}{\sigma^2} d_{t-1} - \sigma_0^2 \Sigma_{>} d_{t-1} \Sigma_{>}^T \right) + \left( U^T y - \Sigma V^T x_{t-1} \right) + \left( V^T \cdot s(x_{t-1}, \sigma_i) \right)$$
10. $$x_t \leftarrow V \cdot \left( V^T x_{t-1} + cA_i d_t + \sqrt{2} \sigma_i z_t \right)$$
11. end
12. $x_0 \leftarrow x_\tau$
13. end

Output: $x_0$

Note that when we set $H = \mathbf{0}$ and $\sigma_0 = 0$, implying no measurements, the above algorithm degenerates to an image synthesis, exactly as in [44]. Two other special cases of this algorithm are obtained for $H = \mathbf{I}$ or $H = \mathbf{I}$ with some rows removed, the first referring to denoising and the second to noisy inpainting, both cases shown in [21]. Lastly, for the choices of $H$ as in [20] or [44, 46] and with $\sigma_0 = 0$, the above algorithm collapses to a close variant of their proposed iterative methods.

5 Experimental Results

In our experiments we use the NCSNv2 [45] network in order to estimate the score function of the prior distribution. Three different NCSNv2 models are used, each trained separately on the training sets of: (i) images of size $64 \times 64$ pixels from the CelebA dataset [27]; (ii) images of size $128 \times 128$ pixels from LSUN [56] bedrooms dataset; and (iii) LSUN $128 \times 128$ images of towers. We demonstrate SNIPS’ capabilities on the respective test sets for image deblurring, super resolution, and compressive sensing. In each of the experiments, we run our algorithm 8 times, producing 8 samples for each input. We examine both the samples themselves and their mean, which serves as an approximation of the MMSE solution, $E[|x|y]$.

Recall that $s(\tilde{x}, \sigma) = (D(\tilde{x}, \sigma) - \tilde{x}) / \sigma^2$, being a denoising residual.
For image deblurring, we use a uniform $5 \times 5$ blur kernel, and an additive white Gaussian noise with $\sigma_0 = 0.1$ (referring to pixel values in the range $[0, 1]$). Figure 1 demonstrates the obtained results for several images taken from the CelebA dataset. As can be seen, SNIPS produces visually pleasing, diverse samples.

For super resolution, the images are downscaled using a block averaging filter, i.e., each non-overlapping block of pixels in the original image is averaged into one pixel in the low-resolution image. We use blocks of size $2 \times 2$ or $4 \times 4$ pixels, and assume the low-resolution image to include an additive white Gaussian noise. We showcase results on LSUN and CelebA in Figures 2, 4, and 5.

For compressive sensing, we use three random projection matrices with singular values of 1, that compress the image by 25%, 12.5%, and 6.25%. As can be seen in Figure 3 and as expected, the more aggressive the compression, the more significant are the variations in reconstruction.

We calculate the average PSNR (peak signal-to-noise ratio) of each of the 8 samples in our experiments, as well as the PSNR of their mean, as shown in Table 1. In all the experiments, the empirical conditional mean presents an improvement of around 2.4 dB in PSNR, even though it is less visually appealing compared to the samples. This is consistent with the theory in [5], which states that the difference in PSNR between posterior samples and the conditional mean (the MMSE estimator) should be 3 dB, with the MMSE estimator having poorer perceptual quality but better PSNR.

A comparison of our deblurring results to those obtained by RED [39] is detailed in the supplemental material. We show that SNIPS exhibits superior performance over RED, achieving more than 11% improvement in PSNR and more than 58% improvement in LPIPS [62], a perceptual quality metric.

5.1 Assessing Faithfulness to the Measurements

A valid solution to an inverse problem should satisfy two conditions: (i) It should be visually pleasing, consistent with the underlying prior distribution of images, and (ii) It should be faithful to the given measurement, maintaining the relationship as given in the problem setting. Since the prior distribution
Table 1: PSNR results for different inverse problems on 8 images from CelebA [27]. We ran SNIPS 8 times, and obtained 8 samples. The average PSNR for each of the samples is in the first column, while the average PSNR for the mean of the 8 samples for each image is in the second one.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Sample PSNR</th>
<th>Mean PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform deblurring</td>
<td>25.54</td>
<td>28.01</td>
</tr>
<tr>
<td>Super resolution (by 2)</td>
<td>25.58</td>
<td>28.03</td>
</tr>
<tr>
<td>Super resolution (by 4)</td>
<td>21.90</td>
<td>24.31</td>
</tr>
<tr>
<td>Compressive sensing (by 25%)</td>
<td>25.68</td>
<td>28.06</td>
</tr>
<tr>
<td>Compressive sensing (by 12.5%)</td>
<td>22.34</td>
<td>24.67</td>
</tr>
</tbody>
</table>

Figure 6: Compressive sensing results on LSUN [56] tower images (compression by 25% and adding noise with $\sigma_0 = 0.04$).

is unknown, we assess the first condition by visually observing the obtained solutions and their tendency to look realistic. As for the second condition, we perform the following computation: We degrade the obtained reconstruction $\hat{x}$ by $H$, and calculate its difference from the given measurement $y$, obtaining $y - H\hat{x}$. According to the problem setting, this difference should be an additive white Gaussian noise vector with a standard deviation of $\sigma_0$. We examine this difference by calculating its empirical standard deviation, and performing the Pearson-D’Agostino [8] test of normality on it, accepting it as a Gaussian vector if the obtained p-value is greater than 0.05. We also calculate the Pearson correlation coefficient (denoted as $\rho$) among neighboring entries, accepting them as uncorrelated for coefficients smaller than 0.1 in absolute value. In all of our tests, the standard deviation matches $\sigma_0$ almost exactly, the Pearson correlation coefficient satisfies $|\rho| < 0.1$, and we obtain p-values greater than 0.05 in around 95% of the samples (across all experiments). These results empirically show that our algorithm produces valid solutions to the given inverse problems.

6 Conclusion and Future Work

SNIPS, presented in this paper, is a novel stochastic algorithm for solving general noisy linear inverse problems. This method is based on annealed Langevin dynamics and Newton’s method, and relies on the availability of a pre-trained Gaussian MMSE denoiser. SNIPS produces a random variety of high quality samples from the posterior distribution of the unknown given the measurements, while guaranteeing their validity with respect to the given data. This algorithm’s derivation includes an intricate choice of the injected annealed noise in the Langevin update equations, and an SVD decomposition of the degradation operator for decoupling the measurements’ dependencies. We demonstrate SNIPS’ success on image deblurring, super resolution, and compressive sensing.

Extensions of this work should focus on SNIPS’ limitations: (i) The need to deploy SVD decomposition of the degradation matrix requires a considerable amount of memory and computations, and hinders the algorithm’s scalability; (ii) The current version of SNIPS does not handle general content images, a fact that is related to the properties of the denoiser being used [41]; and (iii) SNIPS, as any other Langevin based method, requires (too) many iterations (e.g., in our super-resolution tests on CelebA, 2 minutes are required for producing 8 sample images), and means for its acceleration should be explored.
References


