A Massive MIMO Sampling Detection Strategy Based on Denoising Diffusion Model

Lanxin He
College of Electronic and Information Engineering
Nanjing University of Aeronautics and Astronautics, Nanjing, China
Email: lanxin_he@nuaa.edu.cn

Zheng Wang and Yongming Huang
School of Information Science and Engineering
Southeast University, Nanjing, China
Email: wznuaa@gmail.com; huangym@seu.edu.cn

***Abstract*—The Langevin sampling method relies on an accurate score matching while the existing massive multiple-input multiple output (MIMO) Langevin detection involves an inevitable singular value decomposition (SVD) to calculate the posterior score. In this work, a massive MIMO sampling detection strategy that leverages the denoising diffusion model is proposed to narrow the gap between the given iterative detector and the maximum likelihood (ML) detection in an SVD-free manner. Specifically, the proposed score-based sampling detection strategy, denoted as approximate diffusion detection (ADD), is applicable to a wide range of iterative detection methods, and therefore entails a considerable potential in their performance improvement by multiple sampling attempts. On the other hand, the ADD scheme manages to bypass the channel SVD by introducing a reliable iterative detector to produce a sample from the approximate posterior, so that further Langevin sampling is tractable. Customized by the conjugated gradient descent algorithm as an instance, the proposed sampling scheme outperforms the existing score-based detector in terms of a better complexity-performance trade-off.**

***Index Terms*—Massive MIMO detection, diffusion model, denoising score matching, Langevin sampling.**

I. Introduction

The uplink detection in massive multiple-input multiple-output (MIMO) systems is of vital importance for the sake of realizing the substantial benefits of the evolving MIMO techniques [1]. Nevertheless, the computational complexity associated with maximum likelihood (ML) detection is prohibitive for hardware implementation, leading to an urgent request to develop algorithms with more competitive detection trade-off [2]. On the other hand, the score-based generative models, collectively known as *diffusion model*, have lately attracted increasing attention owing to their success in image generation, inpainting, and synthesis [3].

Historically, the original diffusion model is proposed in [4] as an unsupervised generative model. Inspired by non-equilibrium statistical physics, it gradually destroys the objective distribution through a *forward diffusion* process, and then tries to recover this distribution by a *reverse generative* process. Later in [5], a simpler objective function for the denoising diffusion probabilistic model (DDPM) is developed, henceforth simplifying its training and popularizing the diffusion model. What is more, the equivalence between DDPM and the denoising score matching [6] has been shown, which encourages the subsequent unification of DDPM and denoising score matching by stochastic differential equations (SDE) in [7]. Despite the fact that the diffusion model and the score-based generative model are different in the earlier research, the word "diffusion model" now often refers to both of them.

Score matching aims to estimate the *score*, gradient of the logarithm of a density distribution $p(\mathbf{x})$, i.e., $\nabla_{\mathbf{x}} \log p(\mathbf{x})$. It captures the characteristic of the objective distribution and is essential for Langevin dynamics sampling [6]. Given $\mathbf{y}$ as an observed variable, in order to apply the Langevin method in massive MIMO detection problem, one has to find out a way to sample from the posterior distribution $p(\mathbf{x}|\mathbf{y})$ instead of $p(\mathbf{x})$. This can be accomplished by the SNIPS method in [8], where a singular value decomposition (SVD) is required to derive the closed-form expression for posterior score. Afterwards, the first score-based massive MIMO detector [9] successfully samples on the posterior distribution using the SNIPS method and manages to implement a list detection.

In this work, we propose another score-based sampling detection method, namely approximate diffusion detection (ADD), as a more flexible and SVD-free scheme. The main idea is to implement a deterministic detection method stochastically under the denoising diffusion model architecture. Through multiple sampling attempts, the ADD customized by a particular iterative detector, such as the conjugated gradient descent (CGD), is capable of achieving the near-ML performance. Assisted by a reliable iterative detection, the sampling on an approximate posterior $\widehat{p}(\mathbf{x}|\mathbf{y})$ characterized by this given detector is tractable. The technical contribution of this paper is two-fold: We propose a score-based sampling detection strategy that can be flexibly applied to a wide range of detection algorithms, while the proposed scheme is able to achieve a more efficient sampling than the existing score-based detection without channel SVD.

II. Score-Based Massive MIMO Detection

A. System Model

For notational simplicity, we consider the real-valued linear system for massive MIMO detection with K transmitting and N receiving antennas as follows

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{1}$$

Corresponding author: Zheng Wang (email: wznuaa@gmail.com) where the transformation from the complex system model to the real one is straightforward [10]. We assume the flat Rayleigh fading channel matrix $\mathbf{H} \in \mathbb{R}^{N \times K}$ to be perfectly known, and its entries are independent and identically distributed (i.i.d.) with zero mean and unit variance. $\mathbf{y} \in \mathbb{R}^{N}$, $\mathbf{x} \in \mathbb{R}^{K}$ and $\mathbf{n} \in \mathbb{R}^{N}$ denote the transmitted signal, the corresponding received signal and the zero-mean additive white Gaussian noise with variance σ_0^2, respectively. Denote $\mathcal{Q} = \{\pm 1, \pm 3, ..., \pm\sqrt{M} - 1\}$ as the constellation set for M-ary quadrature amplitude modulation (QAM). Given the perfect channel state information (CSI) and an observed $\mathbf{y}$ in (1), the MIMO detection problem aims to find an estimate of $\mathbf{x}$ that maximizes the posterior probability:

$$\begin{aligned} \widehat{\mathbf{x}}_{\text{MAP}} &= \arg\max_{\mathbf{x} \in \mathcal{Q}^K} p(\mathbf{x}|\mathbf{y}, \mathbf{H}) \\ &= \arg\max_{\mathbf{x} \in \mathcal{Q}^K} p(\mathbf{y} - \mathbf{H}\mathbf{x}) p(\mathbf{x}). \end{aligned} \tag{2}$$

As a consequence of uniform assumption on prior $p(\mathbf{x})$ and the Gaussian formulation of the noise $\mathbf{n}$, the optimal maximum a posteriori (MAP) solution would reduce to a maximum likelihood (ML) one, namely

$$\widehat{\mathbf{x}}_{\text{ML}} = \arg\min_{\mathbf{x} \in \mathcal{Q}^K} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2. \tag{3}$$

B. Denoising Score Matching

The *score* of a density distribution $p(\mathbf{x})$ is defined as the gradient of its log-probability, i.e.,

$$\mathbf{s}(\mathbf{x}) \triangleq \nabla_{\mathbf{x}} \log p(\mathbf{x}). \tag{4}$$

The procedure that finds such a function $\mathbf{s}_\theta(\mathbf{x})$ approximating the score is called *score matching* [11], where θ is the parameter either to be fitted in a deep neural network or to be determined by a traditional trace-based method. However, the trace-based methods are not tractable for large-scale systems, and henceforth the denoising score matching [12] methodology is leveraged to bypass the trace calculation.

Specifically, the denoising score matching firstly perturbs the objective distribution $p(\mathbf{x})$ with a predefined Gaussian diffusion kernel $q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x}) = \mathcal{N}(\widetilde{\mathbf{x}}; \mathbf{x}, \sigma^2\mathbf{I})$, leading to a perturbed distribution $q_\sigma(\widetilde{\mathbf{x}}) = \int p(\mathbf{x}) q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x}) d\mathbf{x}$. Here σ controls the similarity between the original distribution and the perturbed one. Then, a score network $\mathbf{s}_\theta(\widetilde{\mathbf{x}})$ is established and optimized to estimate the score of the perturbed distribution $q_\sigma(\widetilde{\mathbf{x}})$ using the following criterion [12]:

$$\theta = \arg\min_{\theta} \mathbb{E}_{q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x})p(\mathbf{x})}[\|\mathbf{s}_\theta(\widetilde{\mathbf{x}}) - \nabla_{\widetilde{\mathbf{x}}} \log q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x})\|^2]. \tag{5}$$

As long as the perturbation is small enough, the optimal network $\mathbf{s}_\theta^*(\mathbf{x})$ satisfies

$$\mathbf{s}_\theta^*(\mathbf{x}) = \nabla_{\mathbf{x}} \log q_\sigma(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p(\mathbf{x}), \tag{6}$$

therefore attaining the score of the objective distribution $p(\mathbf{x})$.

Algorithm 1 Annealed Langevin Sampling (ALS) Detection

```
Input Received signal $\mathbf{y}$, channel matrix $\mathbf{H}$, $\sigma_0$, $T$, $L_A$, noise schedule $\{\sigma_t\}_{t=1}^T$
Calculate the SVD of $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, $\widetilde{\mathbf{x}}_T \sim \mathcal{N}(\mathbf{0}, \sigma_T\mathbf{I})$
for $t = T, \cdots 1$ do
    Update the step size $\delta_t$ according to [9], $\widehat{\mathbf{x}}_0 = \widetilde{\mathbf{x}}_t$
    for $i = 1, \cdots, L_A$ do
        Calculate the posterior score $\mathbf{s}(\mathbf{x}_{i-1}, \sigma_t)$ in [9]
        $\widehat{\mathbf{x}}_i = \widehat{\mathbf{x}}_{i-1} + \frac{\delta_t}{2}\mathbf{s}(\widehat{\mathbf{x}}_{i-1}, \sigma_t) + \sqrt{\delta_t}\mathbf{w}_i$, $\mathbf{w}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    end for
    $\widetilde{\mathbf{x}}_{t-1} = \widehat{\mathbf{x}}_{L_A}$
end for
Output $\widehat{\mathbf{x}} = \widehat{\mathbf{x}}_{L_A}$
```

C. Langevin Dynamics for Massive MIMO Detection

Once the score $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ is obtained, the Langevin method [13] is able to produce samples from the density distribution $p(\mathbf{x})$ given the initial $\widehat{\mathbf{x}}_0$ from a prior distribution π, $\widehat{\mathbf{x}}_0 \sim \pi(\mathbf{x})$, by iterating up to L times as follows,

$$\widehat{\mathbf{x}}_i = \widehat{\mathbf{x}}_{i-1} + \frac{\delta}{2}\nabla_{\mathbf{x}} \log p(\widehat{\mathbf{x}}_{i-1}) + \sqrt{\delta}\mathbf{w}_i, \ i = 1, 2, \cdots, L. \tag{7}$$

Here $\mathbf{w}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, i denotes the sampling time step and δ is the sampling step size. However, the naive application of Langevin method would encounter irregular fluctuation, estimation inaccuracy as well as slow mixing problem. Henceforth the *annealed Langevin* strategy that adopts multiple noise levels is preferred to obviate these difficulties [6]. Specifically, a noise schedule $\{\sigma_t\}_{t=1}^T$ that satisfies $\frac{\sigma_1}{\sigma_2} = \cdots = \frac{\sigma_{T-1}}{\sigma_T} < 1$ divides the whole perturbation into several intermediate *forward diffusion* processes $q_{\sigma_t}(\widetilde{\mathbf{x}}_t|\mathbf{x}) \sim \mathcal{N}(\widetilde{\mathbf{x}}_t; \mathbf{x}, \sigma_t^2\mathbf{I})$. This can be re-parameterized as the following Markov chain:

$$\widetilde{\mathbf{x}}_t = \widetilde{\mathbf{x}}_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2}\mathbf{z}_t, \quad t = 1, \cdots, T, \tag{8}$$

where $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a Gaussian random noise and $\widetilde{\mathbf{x}}_0 \triangleq \mathbf{x}$.

Then, an effort to recover these perturbed $\{\widetilde{\mathbf{x}}_t\}_{t=1}^T$ reversely is made in the *reverse generative* process, where a noise conditional score network $\mathbf{s}_\theta(\mathbf{x}, \sigma_t)$ is involved to train. Once the training is done, a successive L_A-times sampling is conducted along $q_{\sigma_t}(\widetilde{\mathbf{x}}_t)$ in a descending order, i.e., $\widetilde{\mathbf{x}}_T, \cdots, \widetilde{\mathbf{x}}_1$. To be more specific, for each L_A-times sampling, it starts from $\widehat{\mathbf{x}}_0 = \widetilde{\mathbf{x}}_t$, and the final sample $\widehat{\mathbf{x}}_{L_A}$ is treated as an estimate of perturbed variable for the next iteration, namely $\widetilde{\mathbf{x}}_{t-1} = \widehat{\mathbf{x}}_{L_A}$, as follows,

$$\widehat{\mathbf{x}}_i = \widehat{\mathbf{x}}_{i-1} + \frac{\delta_t}{2}\mathbf{s}_\theta(\widehat{\mathbf{x}}_{i-1}, \sigma_t) + \sqrt{\delta_t}\mathbf{w}_i, i = 1, 2, \cdots, L_A. \tag{9}$$

Here $\delta_t = \varepsilon \cdot \sigma_t^2/\sigma_T^2$, with ε being the annealed learning rate, updates corresponding to σ_t in a reverse order, i.e., from σ_T to σ_1. Clearly, this annealed Langevin strategy requires $L_A \times T$ times sampling in total to produce a single reliable sample, and generally $L_A \times T \leq L$ stands as a result of the improved mixing rate of annealed Langevin dynamics.

Nevertheless, the detection problem needs to tackle the issue of sampling on the posterior density distribution $p(\mathbf{x}|\mathbf{y})$ rather

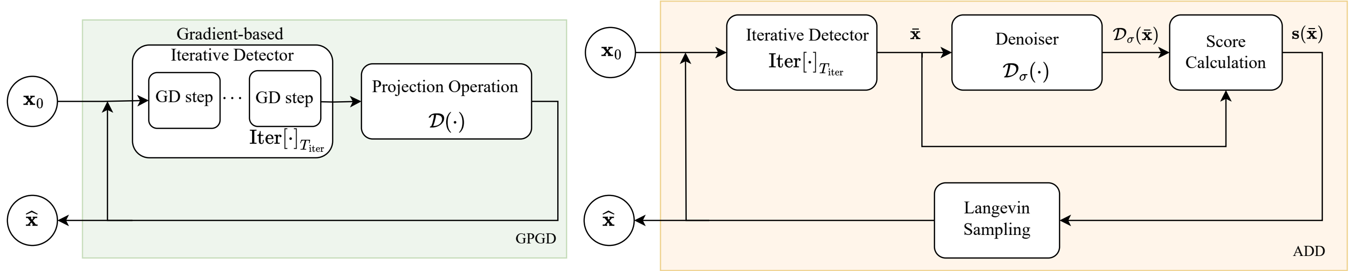

Fig. 1. Framework of GPGD and the proposed ADD sampling scheme.

than $p(\mathbf{x})$. One might handle this by applying the Bayes' rule and rewriting the posterior score as

$$\nabla_{\widehat{\mathbf{x}}_t} \log p(\widehat{\mathbf{x}}_t|\mathbf{y}) = \nabla_{\widehat{\mathbf{x}}_t} \log p(\widehat{\mathbf{x}}_t) + \nabla_{\widehat{\mathbf{x}}_t} \log p(\mathbf{y}|\widehat{\mathbf{x}}_t), \quad (10)$$

whereas a singular value decomposition (SVD) of the channel matrix $\mathbf{H}$ is required to formulate this posterior score explicitly, thereby imposing strains on computation complexity for large-scale systems. This annealed Langevin sampling (ALS) detection [9] is outlined in Alg.1 for one single sampling trajectory. Finally, the entire Langevin dynamics produces an S-length sample list $\mathcal{L} = \{\widehat{\mathbf{x}}^{(1)}, \widehat{\mathbf{x}}^{(2)}, \cdots, \widehat{\mathbf{x}}^{(S)} | \widehat{\mathbf{x}} \sim p(\mathbf{x}|\mathbf{y})\}$ for detection, amongst which the sample that minimizes following Euclidean distance is distinguished as the final estimate:

$$\widehat{\mathbf{x}} = \underset{\mathbf{x} \in \mathcal{L}}{\arg\min} \, \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2. \quad (11)$$

III. Approximate Diffusion Detection Strategy

In this section, we present the proposed ADD methodology in detail. We mention that the neural network training is not necessary here, but ADD is still amenable to further deep generative detection network extension. Meanwhile, it is shown by [7] that the Markov chain in (8) is equivalent to the following SDE:

$$\mathrm{d}\mathbf{x} = \sqrt{\frac{\mathrm{d}\left[\sigma^2(t)\right]}{\mathrm{d}t}} \mathrm{d}\boldsymbol{\omega}, \quad (12)$$

where $\boldsymbol{\omega}$ is the standard Wiener process, and its diffusion coefficient is $g(t) = \sqrt{\frac{\mathrm{d}[\sigma^2(t)]}{\mathrm{d}t}}$. Basically, this SDE assumes that, as $T \to \infty$, the Markov chain $\{\widetilde{\mathbf{x}}_t\}_{t=1}^T$ turns into a continuous stochastic process $\{\widetilde{\mathbf{x}}(t)\}_{t=0}^1$, where the integer index $t = 1, \cdots, T$ is substituted by a continuous time variable $t \in [0, 1]$. For the same reason, the noise schedule $\{\sigma_t\}_{t=1}^T$ alters to a function $\sigma(t)$ and $\mathbf{z}_t$ becomes a Gaussian process $\mathbf{z}(t)$. Accordingly, its reverse SDE is given by

$$\mathrm{d}\mathbf{x} = -g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) \mathrm{d}t + g(t) \mathrm{d}\bar{\boldsymbol{\omega}}, \quad (13)$$

where $\bar{\boldsymbol{\omega}}$ is the Wiener process through a reverse time and $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ is the score at time t. The SDE-manner formulation unifies both the denoising score matching and the diffusion model, and therefore is adopted here for further convenience. Note that for the forward process in (12), the time variable t evolves from $t = 0$ to $t = 1$. By contrast, it reduces from $t = 1$ to $t = 0$ in the reverse process of (13), which accounts for distinguishing this reverse generating from the forward diffusion process.

A. Framework

The proposed ADD strategy is inspired by the generalized projected gradient descent (GPGD) method in [10], where a projection operator alternates with an iterative gradient-based detector, but the proposed strategy is also applicable to other reliable detection algorithms. Besides, the ADD manages to circumvent the calculation of posterior score by SVD and results in a computational efficiency.

Typically, with respect to the GPGD method, a gradient-based iterative detector, denoted by $\mathrm{Iter}\,[\cdot]_{T_{\mathrm{iter}}}$, implements T_{iter} times gradient descent (GD) steps to provide an estimate of the signal. Afterwards a projection operator $\mathcal{D}$ tries to find the ideal counterpart of this estimate in a specific discrete domain. One iteration of this process can be expressed as

$$\mathbf{x}_{i+1} = \mathcal{D}\left(\mathrm{Iter}\,[\mathbf{x}_i, \mathbf{y}; \mathbf{H}]_{T_{\mathrm{iter}}}\right), \quad (14)$$

with i being the iteration index. In practical, the projection operator $\mathcal{D}$ might be constructed by a deep neural network and trained under a minimum mean square error (MSE) criterion, but finding a good enough projection is always time-consuming and requires dedicated network design. The main idea behind the proposed strategy is to transform the deterministic algorithm in a stochastic manner, where the performance gain can be attained by multiple sampling attempts, thus shrinking the gap to the optimal ML detection.

In particular, during the ADD, an iterative method is utilized to produce a relatively reliable solution $\bar{\mathbf{x}} = \mathrm{Iter}[\mathbf{x}]_{T_{\mathrm{iter}}}$ that supports the subsequent score estimation. This variable characterizes the distribution of an approximate posterior, denoted by $\widehat{p}(\mathbf{x}|\mathbf{y})$, which is the key to eliminate the closed-form score calculation. Based on this, a denoiser $\mathcal{D}_\sigma(\cdot)$ parameterized by σ, plays a similar role as the projection operator in GPGD method to give a minimum MSE estimate $\mathcal{D}_\sigma(\bar{\mathbf{x}})$. By doing so, the score of approximate posterior can be given by the Tweedie's identity, whose feasibility can be found in [14]:

$$\mathbf{s}(\bar{\mathbf{x}}) = \frac{\mathcal{D}(\bar{\mathbf{x}}) - \bar{\mathbf{x}}}{\sigma^2}. \quad (15)$$

Leveraging this score, the Langevin sampling can be conducted to sample from the approximate posterior $\widehat{p}(\mathbf{x}|\mathbf{y})$. The proposed ADD strategy and the GPGD method are compared in Fig.1, where their related structures can be observed. Since the denoiser in ADD works as an MSE minimizer, which is often the role played by the projection in GPGD method, extra projection in the ADD is not necessary.

Algorithm 2 Approximate Diffusion Detection (ADD)

```
Input Received signal $\mathbf{y}$, channel matrix $\mathbf{H}$, symbol energy $E_s$, $\sigma_0$, $\sigma_{\min}$, $\sigma_{\max}$, $\epsilon$, $T$, $T_{\text{iter}}$, $S$
Calculate the noise schedule $\sigma(t) = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^t$ and the diffusion coefficient $g(t) = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^t\sqrt{2\log\frac{\sigma_{\max}}{\sigma_{\min}}}$ for $t \in [\epsilon, 1]$ with the interval $\Delta t = \frac{1-\epsilon}{T}$
for $j = 1, ..., S$ do
    $t = 1$, $\widetilde{\mathbf{y}}(0) = \mathbf{y}$, $\widehat{\mathbf{x}}(1) \sim \mathcal{N}(\mathbf{0}, \sigma_{\max}^2\mathbf{I})$
    while $t \neq \epsilon$ do
        Sample $\widetilde{\mathbf{y}}(t) \sim \mathcal{N}\left(\widetilde{\mathbf{y}}(t); \mathbf{y}(0), \sigma(t)^2\mathbf{H}\mathbf{H}^T\right)$
        $t = t - \Delta t$
        Evoke Function 1 to get $\bar{\mathbf{x}}(t)$
        Estimate the score $\mathbf{s}\big(\bar{\mathbf{x}}(t)\big) = \dfrac{\mathcal{D}_{\sigma(t)}(\bar{\mathbf{x}}(t)) - \bar{\mathbf{x}}(t)}{\sigma(t)^2}$, where $\mathcal{D}_{\sigma(t)}(\cdot)$ adopts the LGD form in (22)
        Sample $\mathbf{w}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and get
            $\widehat{\mathbf{x}}(t) = \bar{\mathbf{x}}(t) - g^2(t)\cdot\mathbf{s}\big(\bar{\mathbf{x}}(t)\big)\Delta t + g(t)\sqrt{\Delta t}\cdot\mathbf{w}(t)$
    end while
    Include $\widehat{\mathbf{x}}(\epsilon)$ to the list $\mathcal{L}$ of length $S$.
end for
Output $\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathcal{L}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$.
```

Function 1 Conjugate Gradient Descent (CGD)

```
Input $\mathbf{y} = \widetilde{\mathbf{y}}(t)$, $\mathbf{x}_0 = \widehat{\mathbf{x}}(t)$, $\mathbf{H}$, $T_{\text{iter}}$
Initialization: $\mathbf{A} = \mathbf{H}^T\mathbf{H} + \sigma_0 E_s\mathbf{I}$, $\mathbf{b} = \mathbf{H}^T\mathbf{y}$
    $\mathbf{r}_0 = \mathbf{A}\mathbf{x}_0 - \mathbf{b}$, $\mathbf{d}_0 = -\mathbf{r}$
for $i = 0, 1, \cdots, T_{\text{iter}} - 1$ do
    $\alpha_{i+1} = (\mathbf{r}_i^T\mathbf{r}_i)/(\mathbf{r}_i^T\mathbf{A}\mathbf{d}_i)$
    $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_{i+1}\mathbf{d}_i$
    $\mathbf{r}_{i+1} = \mathbf{r}_i + \alpha_{i+1}\mathbf{A}\mathbf{d}_i$
    $\beta_{i+1} = (\mathbf{r}_{i+1}^T\mathbf{r}_{i+1})/(\mathbf{r}_i^T\mathbf{r}_i)$
    $\mathbf{d}_{i+1} = -\mathbf{r}_{i+1} + \beta_{i+1}\mathbf{d}_i$
end for
Return $\mathbf{x}_{T_{\text{iter}}}$
```

B. Implementation

The implementation details of the proposed ADD method are elaborated in the following. Here we define the noise schedule function as

$$\sigma(t) = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^t, \quad t \in [0, 1], \tag{16}$$

with $\sigma(0) = \sigma_{\min} \triangleq \sigma_1$ and $\sigma(1) = \sigma_{\max} \triangleq \sigma_T$. We still denote T as the total sampling times, and henceforth the interval between each individual time step is $\Delta t = \frac{1}{T}$.

In this way, the one-step transition probability of the forward diffusion process can be written as

$$\begin{aligned} &p\left(\widetilde{\mathbf{x}}(t)|\widetilde{\mathbf{x}}(t-\Delta t)\right) \\ &= \mathcal{N}\big(\widetilde{\mathbf{x}}(t); \widetilde{\mathbf{x}}(t-\Delta t), \big(\sigma^2(t) - \sigma^2(t-\Delta t)\big)\mathbf{I}\big), t \in [0, 1], \end{aligned} \tag{17}$$

and the corresponding t-step counterpart is

$$p\left(\widetilde{\mathbf{x}}(t)|\mathbf{x}(0)\right) = \mathcal{N}\left(\widetilde{\mathbf{x}}(t); \mathbf{x}(0), \sigma(t)^2\mathbf{I}\right). \tag{18}$$

According to this, the perturbed variable $\widetilde{\mathbf{x}}(t)$ in the forward diffusion is re-parametrized as $\widetilde{\mathbf{x}}(t) = \mathbf{x}(0) + \sigma(t)\mathbf{z}(t)$. Therefore, the perturbed received signal at time t can be formulated as

$$\begin{aligned} \widetilde{\mathbf{y}}(t) &= \mathbf{H}\widetilde{\mathbf{x}}(t) + \mathbf{n} = \mathbf{H}\big(\mathbf{x}(0) + \sigma(t)\mathbf{z}(t)\big) + \mathbf{n} \\ &= \mathbf{H}\mathbf{x}(0) + \mathbf{n} + \sigma(t)\mathbf{H}\mathbf{z}(t) = \mathbf{y} + \sigma(t)\mathbf{H}\mathbf{z}(t) \end{aligned} \tag{19}$$

with $\mathbf{x}(0) = \mathbf{x}$. Now the t-step transition probability of $\widetilde{\mathbf{y}}(t)$ is also attainable:

$$p\left(\widetilde{\mathbf{y}}(t) \mid \mathbf{y}(0)\right) \sim \mathcal{N}\left(\widetilde{\mathbf{y}}(t); \mathbf{y}(0), \sigma(t)^2\mathbf{H}\mathbf{H}^T\right), \tag{20}$$

where $\mathbf{y}(0) = \mathbf{y}$. In this way, sampling on the perturbed received signal $\widetilde{\mathbf{y}}(t)$ is tractable. Hence, starting from the time $t = 1$, the reverse sample $\widehat{\mathbf{x}}(t)$ and forward $\widetilde{\mathbf{y}}(t)$ can cooperate to give an estimate $\bar{\mathbf{x}}(t)$ by the iterative detection method, i.e.,

$$\bar{\mathbf{x}}(t) = \text{Iter}\left[\widehat{\mathbf{x}}(t), \widetilde{\mathbf{y}}(t); \mathbf{H}\right]_{T_{\text{iter}}}. \tag{21}$$

This estimate is viewed as a sample from the approximate posterior $\widehat{p}(\mathbf{x}|\mathbf{y})$ depending on the chosen iterative detection method.

Now that the variable $\bar{\mathbf{x}}$ encapsulates information about $\mathbf{y}$ and $\mathbf{H}$, once a minimum MSE estimate of $\bar{\mathbf{x}}$ is obtained, the score of $\widehat{p}(\mathbf{x}|\mathbf{y})$ can be analyzed according to (15): $\mathbf{s}\big(\bar{\mathbf{x}}(t)\big) = \dfrac{\mathcal{D}_{\sigma(t)}(\bar{\mathbf{x}}(t)) - \bar{\mathbf{x}}(t)}{\sigma(t)^2}$. The required denoiser $\mathcal{D}_{\sigma(t)}$ can be established by a deep neural network, while for now we directly use the training-free estimator as in [9]. Specifically, this estimator evaluates every component x_k of the input $\mathbf{x}$ aided by the one-dimensional (1-D) lattice Gaussian distribution (LGD), namely

$$p_{\mathcal{Q}}(x_k = \widehat{x}_k; \bar{x}_k, \sigma) \triangleq \frac{1}{Z_{\mathcal{Q}}}\exp\left(\frac{-\|\widehat{x}_k - \bar{x}_k\|^2}{2\sigma^2}\right), \widehat{x}_k \in \mathcal{Q}, \tag{22}$$

with $Z_{\mathcal{Q}} = \sum_{\widehat{x}_k\in\mathcal{Q}}\exp\left(\frac{-\|\widehat{x}_k - \bar{x}_k\|^2}{2\sigma^2}\right)$ a normalization scalar.

To this end, we can generate a new sample $\widehat{\mathbf{x}}$ utilizing the attained score as (13) indicates, leading to

$$\widehat{\mathbf{x}}(t) = \bar{\mathbf{x}}(t) - g^2(t)\cdot\mathbf{s}\big(\bar{\mathbf{x}}(t)\big)\Delta t + g(t)\sqrt{\Delta t}\cdot\mathbf{w}(t), \tag{23}$$

where $\mathbf{w}(t)$ is the Gaussian process from $\mathbf{w}_t$. As the sampling goes on from $t = 1$ to $t = 0$, the final sample $\widehat{\mathbf{x}}(0)$ is treated as sampled from the approximate posterior distribution $\widehat{p}(\mathbf{x}|\mathbf{y})$. The whole generative procedure is illustrated in Fig.2. This completes a single trajectory for the sampling, and this procedure continues up to S times, generating a candidate list for final decision. The overall algorithm is outlined in Alg.2, where the CGD method [15], as presented in **Func.1**, is chosen to customize the Iter $[\cdot]_{T_{\text{iter}}}$ procedure as a simple instance. It is suggested that $\sigma_{\min}$ should be set as a very small value, like $\sigma_{\min} = 0.01$, and $\sigma_{\max}$ close to the symbol energy E_s. Meanwhile, a extremely small-valued $\epsilon > 0$ is recommended for numerical stability.

C. Complexity Analysis

For one single trajectory, the computation for ADD strategy mainly consists of three parts: selected iterative detector, score calculation and sampling. Particularly, with respect to the sampling, the generation of $\widetilde{\mathbf{y}}(t) \sim \mathcal{N}\left(\widetilde{\mathbf{y}}(t); \mathbf{y}(0), \sigma(t)^2\mathbf{H}\mathbf{H}^T\right)$ is achieved by re-parameterizing $\widetilde{\mathbf{y}}(t) = \mathbf{y} + \sigma(t)\mathbf{H}\mathbf{z}(t)$, which involves multiplying $\mathbf{H}$ to a noise vector $\mathbf{z}(t)$ with NK multiplications. Besides, there are $2T$ times random Gaussian noise generation, one $\mathbf{z}(t)$ and one $\mathbf{w}(t)$ for each iteration, but their complexity can be neglected. Moreover, since the denoiser $\mathcal{D}_\sigma(\cdot)$ adopted in this work evaluates by 1-D LGD, whose complexity is $O(MK)$ for M-QAM, the score calculation part does not require any neural network computing. Finally, as for the iterative detector, taking CGD as an example, its complexity is dominated by calculation of $\mathbf{H}^T\mathbf{H}$, which is of order $O(NK^2)$, and the complexity for one single iterative step is $O(K^2)$. Therefore, the overall complexity of the proposed ADD strategy is $O(NK^2 + T(NK + T_{\text{iter}}K^2 + MK))$ in a polynomial manner. Compared to the ALS method, whose complexity is $O(NK^2 + L_AT(K^2 + MK))$ [9] and largely affected by the SVD operation, the dominant computation of ADD is the chosen iterative detector. Here for the CGD case, the calculation of Hermitian matrix $\mathbf{H}^T\mathbf{H}$ is much simpler than SVD computation, and additionally ADD provides a more flexible scheme with a substitutable iterative detector, leading to a possibility in future complexity-reduction.

D. Discussions On Deep Learning Extension

In the following, we show that further deep learning extension is possible in the proposed ADD scheme. Consider the diffusion kernel $q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x}) = \mathcal{N}(\widetilde{\mathbf{x}}; \mathbf{x}, \sigma^2 I)$ in the mentioned denoising score matching. It is easy to derive that its score has the following form:

$$\nabla_{\widetilde{\mathbf{x}}} \log q_\sigma(\widetilde{\mathbf{x}} \mid \mathbf{x}) = -(\widetilde{\mathbf{x}} - \mathbf{x})/\sigma^2. \tag{24}$$

According to the objective function in (5), the training of a denoising score network $\mathbf{s}_\theta(\widetilde{\mathbf{x}}, \sigma)$ turns to minimize

$$\ell(\theta; \sigma) = \mathbb{E}_{p(\mathbf{x})}\mathbb{E}_{\widetilde{\mathbf{x}} \sim q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x})}\left[\left\|\mathbf{s}_\theta(\widetilde{\mathbf{x}}, \sigma) + \frac{\widetilde{\mathbf{x}} - \mathbf{x}}{\sigma^2}\right\|^2\right]. \tag{25}$$

Again, $\widetilde{\mathbf{x}}$ can be attained by the re-parameterization $\widetilde{\mathbf{x}} = \mathbf{x} + \sigma\mathbf{z}$ with $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Henceforth, by substituting $(\widetilde{\mathbf{x}} - \mathbf{x})/\sigma = \mathbf{z}$, the loss function in (25) can be transformed into

$$\ell(\theta; \sigma) = \mathbb{E}_{p(\mathbf{x})}\mathbb{E}_{\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}\left[\left\|\mathbf{s}_\theta(\widetilde{\mathbf{x}}, \sigma) + \frac{\mathbf{z}}{\sigma}\right\|^2\right]. \tag{26}$$

This corresponds to the *unsupervised* training scheme used in diffusion generative model.

With the Tweedie's identity, the score of a Gaussian perturbed variable is estimated by a minimum MSE denoiser $\mathcal{D}_\sigma(\cdot)$ to get $\mathbf{s}(\widetilde{\mathbf{x}}, \sigma) = \dfrac{\mathcal{D}_\sigma(\widetilde{\mathbf{x}}) - \widetilde{\mathbf{x}}}{\sigma^2}$, as in the proposed ADD scheme. Now if we directly use this score to replace the trainable score network $\mathbf{s}_\theta(\widetilde{\mathbf{x}}, \sigma)$ in (25), the loss function now can be estimated as

$$\ell(\theta; \sigma) = \mathbb{E}_{p(\mathbf{x})}\mathbb{E}_{\widetilde{\mathbf{x}} \sim q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x})}\left[\|\mathcal{D}_{\theta,\sigma}(\widetilde{\mathbf{x}}) - \mathbf{x}\|^2\right], \tag{27}$$

where a subscript θ that represents the trainable parameter is added to the denoiser. By doing so, the problem of training a score network $\mathbf{s}_\theta(\widetilde{\mathbf{x}}, \sigma)$ becomes finding a denoising network $\mathcal{D}_{\theta,\sigma}(\widetilde{\mathbf{x}})$ that minimizes the loss function (27), which turns the unsupervised training into a *supervised* one. The labels are original data $\mathbf{x}$ and an ideal $\mathcal{D}_{\theta,\sigma}(\widetilde{\mathbf{x}})$ shall be able to recover $\mathbf{x}$ given the perturbed data $\widetilde{\mathbf{x}}$ diffused by $q_\sigma(\widetilde{\mathbf{x}}|\mathbf{x})$. Some inspirations might be found in the denoising auto-encoder (DAE) architecture [16]. On the other hand, based on (27), the 1-D LGD denoiser used in ADD inevitably results in a considerable quantization error in the beginning of sampling, where the noise setting $\sigma(t)$ is relatively large. Henceforth, there is a potential performance gain in the ADD scheme to be realized by means of deep learning, leading to a deep generative detection network.

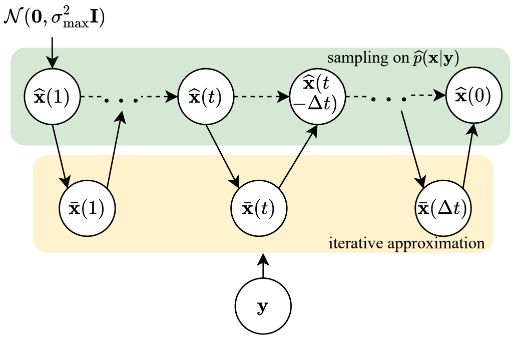
$\mathcal{N}(\mathbf{0}, \sigma^2_{\max}\mathbf{I})$
sampling on $\widehat{p}(\mathbf{x}|\mathbf{y})$
$\widehat{\mathbf{x}}(1)$
$\widehat{\mathbf{x}}(t)$
$\widehat{\mathbf{x}}(t - \Delta t)$
$\widehat{\mathbf{x}}(0)$
$\bar{\mathbf{x}}(1)$
$\bar{\mathbf{x}}(t)$
$\bar{\mathbf{x}}(\Delta t)$
iterative approximation
$\mathbf{y}$

Fig. 2. The reverse generative process in the proposed ADD strategy.

IV. Simulations

This section examines the performance of ADD strategy, where the CGD is selected to be the inner iterative detector and perfect CSI is assumed at the receiver. For comparison purposes, the performances of ALS, minimum mean square error (MMSE) , MMSE-based successive interference cancellation (MMSE-SIC) and ML detection are also shown.

Fig.3 shows the bit error rate (BER) for an uncoded system with $N = K = 32$ and 4-QAM. For fair comparison, we first consider $T \times L_A = T \times T_{\text{iter}}$ case, where ADD and ALS can be taken as implementing for the same iterations. Clearly, with 5 calls, the performance of ADD is more satisfying than that of ALS, and by increasing the sampling trajectories, further performance improvement can be observed by both of the sampling detector. The reason about this gap between ADD and ALS might be the intrinsic property of SNIPS method, where the diffusion noise is assumed in a special way to promise the dependence on the channel noise, but this assumption might be less reliable in the low signal-noise-ratio region. An interesting thing about the ADD is, that the performance can be even enhanced with a fewer T_{iter} (See ADD-20, $T_{\text{iter}} = 3$). This is similar to the phenomenon found in GPGD scheme [10], claiming that there exists the most suitable T_{iter} for particular linear iterative method. Besides, after increasing the trajectories of ADD to 50, only a small gain can be observed, which implies that the convergence of ADD is rather fast but its performance upper bound still gets a nearly 2dB gap to

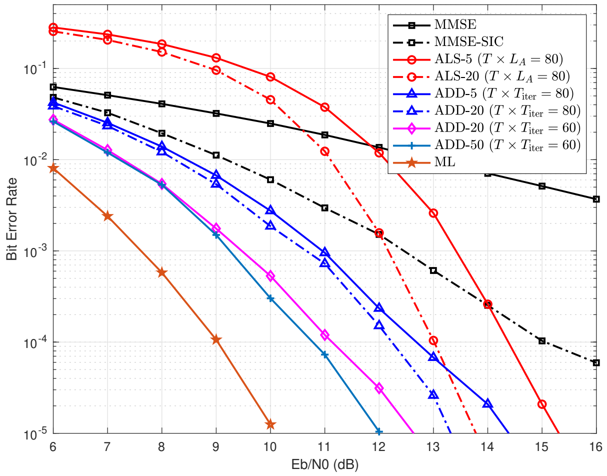

Fig. 3. Performance comparison between the proposed ADD and ALS method for 32×32 system using 4-QAM.

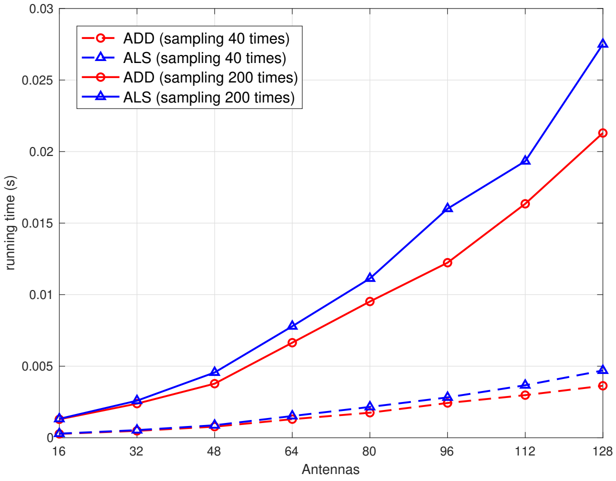

Fig. 4. Running time of ADD and ALS for one trajectory ($N = K$).

the ML detection. Recall that this ADD instance just adopts a simple linear iterative detector, which could have eventually converged to the linear MMSE detection. Nevertheless, it still manages to significantly outperform the nonlinear MMSE-SIC and partially recover the gap to ML detection.

Fig.4 compares the average running time of ADD and ALS for one single trajectory. The ADD is configured as $T_{\text{iter}} = 1$. It can be seen that these two methods show a similar increasing tendency with the dimension, while the ADD still slightly enjoys a more competitive time complexity for the absence of SVD operation. Besides, as shown in Fig.3, the ADD outperforms ALS under the same configuration, especially at low signal-noise-ratio regime, thereby achieving a better complexity-performance trade-off.

V. Conclusion

In this paper, we proposed an approximate diffusion detection (ADD) strategy for massive MIMO systems, which can be applied to a wide range of iterative detectors and eliminates the SVD in existing score-based method. The resultant ADD turned a deterministic iterative detector into a stochastic one to achieve performance gain through sampling. This introduced inner iterative detector helped ADD produce samples based on an approximate posterior, thereby circumventing the SVD required for explicit score calculation. The conducted simulations verified that a fraction of the gap to ML detection can be recovered by the adoption of ADD scheme. Besides, the presented ADD customized by a CGD detector enjoys a more satisfying complexity-performance trade-off compared to the existing score-based detector, but more attempts on this replaceable detector are encouraged. Besides, we discussed the possibility of extending the ADD scheme to a deep generative detection network, which constitutes our future work.

VI. Acknowledgment

This work was supported in part by National Natural Science Foundation of China under Grants No. 62371124, and in part by the National Key R&D Program of China under Grants No. 2023YFC2205501.

References

[1] S. Yang and L. Hanzo, “Fifty years of MIMO detection: the road to large-scale MIMOs,” *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 2015.

[2] Y. Wang et al., “Transformer-empowered 6G intelligent networks: from massive MIMO processing to semantic communication,” *IEEE Trans. Wireless Commun.*, vol. 30, no. 6, pp. 127-135, December 2023.

[3] L. Yang et al., “Diffusion models: a comprehensive survey of methods and applications,” *ACM Compt. Surv.*, vol. 56, no. 105, pp. 1–39, 2023.

[4] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 37, 2015, pp. 2256–2265.

[5] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 33, 2020, pp. 6840–6851.

[6] Y. Song and S. Ermon, “Generative modeling by estimating gradients of the data distribution,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2019, pp. 11918–11930.

[7] Y. Song *et al.*, “Score-based generative modeling through stochastic differential equations,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

[8] B. Kawar, G. Vaksman, and M. Elad, “SNIPS: solving noisy inverse problems stochastically,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 34, 2021, pp. 21757–21769.

[9] N. Zilberstein, C. Dick, R. Doost-Mohammady, A. Sabharwal, and S. Segarra, “Annealed Langevin dynamics for massive MIMO detection,” *IEEE Trans. Wireless Commun.*, vol. 22, no. 6, pp. 3762-3776, 2023.

[10] L. He, Z. Wang, S. Yang, T. Liu and Y. Huang, “Generalizing projected gradient descent for deep-learning-aided massive MIMO detection,” *IEEE Trans. Wireless Commun.*, vol. 23, no. 3, pp. 1827-1839, 2024.

[11] A. Hyvärinen, “Estimation of non-normalized statistical models by score matching,” *J. Mach. Learn. Res.*, vol. 6, pp. 695–709, 2005.

[12] P. Vincent, “A connection between score matching and denoising autoencoders,” *Neural Comput.*, vol. 23, no. 7, pp. 1661–1674, 2011.

[13] A. Barbu and S. Zhu, “Hamiltonian and Langevin Monte Carlo” in Monte Carlo Methods, Springer, pp. 281–325, 2020.

[14] K. Zahra and S. Eero, “Stochastic solutions for linear inverse problems using the prior implicit in a denoiser,” in *Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 34, 2021, pp. 13242–13254.

[15] S. Wright *et al.*, Numerical Optimization, Springer, 1999.

[16] P. Vincent, H. Larochelle, Y. Bengio and P. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1096–1103.