



OPEN ACCESS

EDITED BY

Nicholas Chancellor,
Durham University, United Kingdom

REVIEWED BY

Catherine Potts,
D-Wave Systems, Canada
Teng Bian,
Facebook, United States

*CORRESPONDENCE

Phillip Kerger
✉ pkerger@jhu.edu

RECEIVED 21 August 2023

ACCEPTED 02 October 2023

PUBLISHED 19 October 2023

CITATION

Kerger P and Miyazaki R (2023) Quantum image denoising: a framework via Boltzmann machines, QUBO, and quantum annealing. *Front. Comput. Sci.* 5:1281100. doi: 10.3389/fcomp.2023.1281100

COPYRIGHT

© 2023 Kerger and Miyazaki. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Quantum image denoising: a framework via Boltzmann machines, QUBO, and quantum annealing

Phillip Kerger^{1,2,3*} and Ryoji Miyazaki^{4,5}

¹Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD, United States, ²Quantum Artificial Intelligence Laboratory, NASA Ames Research Center, Moffett Field, CA, United States, ³Research Institute of Advanced Computer Science, Universities Space Research Association (USRA), Moffett Field, CA, United States, ⁴Secure System Platform Research Laboratories, NEC Corporation, Kawasaki, Japan, ⁵NEC-AIST Quantum Technology Cooperative Research Laboratory, National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan

We investigate a framework for binary image denoising via restricted Boltzmann machines (RBMs) that introduces a denoising objective in quadratic unconstrained binary optimization (QUBO) form well-suited for quantum annealing. The denoising objective is attained by balancing the distribution learned by a trained RBM with a penalty term for derivations from the noisy image. We derive the statistically optimal choice of the penalty parameter assuming the target distribution has been well-approximated, and further suggest an empirically supported modification to make the method robust to that idealistic assumption. We also show under additional assumptions that the denoised images attained by our method are, in expectation, strictly closer to the noise-free images than the noisy images are. While we frame the model as an image denoising model, it can be applied to any binary data. As the QUBO formulation is well-suited for implementation on quantum annealers, we test the model on a D-Wave Advantage machine, and also test on data too large for current quantum annealers by approximating QUBO solutions through classical heuristics.

KEYWORDS

denoising, quantum annealing, machine learning, image processing, quadratic unconstrained binary optimization

1. Introduction

Quantum annealing (QA) (Kadowaki and Nishimori, 1998; Das and Chakrabarti, 2008; Albash and Lidar, 2018) is a promising technology for obtaining good solutions to difficult optimization problems, by making use of quantum interactions to aim to solve Ising or quadratic unconstrained binary optimization (QUBO) instances. Since Ising and QUBO instances are NP-hard, and many other combinatorial optimization problems can be reformulated as Ising or QUBO instances (see e.g., Glover et al., 2018), QA has the potential to become an extremely useful tool for optimization. As the capacities of commercially available quantum annealers continue to improve rapidly, it is of great interest to build models that are well-suited for this emerging technology. Furthermore, QA has promising machine learning applications surrounding Boltzmann Machines (BMs), as both QA and BMs are closely connected to the Boltzmann distribution. Boltzmann Machines are a type of generative artificial neural network that aim to learn the distribution of some training data set by fitting a Boltzmann distribution to the data, as described thoroughly in (Goodfellow et al., 2016, §20). On the other hand, QA aims to produce approximate minimum energy

(maximum likelihood) solutions to a Boltzmann distribution via finding the ground state of the associated Hamiltonian that determines the distribution. Hence, maximum likelihood type problems on BMs are a natural candidate for applying QA in a machine learning framework. We contribute to the goal of furthering useful applications of QA in machine learning in this paper by building an image denoising model particularly well-suited for implementation via QA.

The task of image denoising is a fundamental problem in image processing and machine learning. In any means of collecting images, there is always a chance of some pixels being afflicted by noise that we wish to remove; see e.g., [Boyat and Joshi \(2015\)](#) for a good overview. Accordingly, many classical and data-driven approaches to the image denoising problem have been studied in the literature ([Greig et al., 1989](#); [Rudin et al., 1992](#); [Buades et al., 2005](#); [Tang et al., 2012](#); [Cho, 2013](#)). This paper studies a quantum binary image denoising model using Restricted Boltzmann Machines (RBMs henceforth) ([Goodfellow et al., 2016](#), §20.2) that can take advantage of QA by formulating the denoising problem as a QUBO instance. Specifically, given a trained RBM, we introduce a penalty-based denoising scheme that admits a simple QUBO form, for which we derive the statistically optimal penalty parameter as well as a practically-motivated robustness modification. The denoising step only needs to solve a QUBO admitting a bipartite graph representation, and so is well-suited for QA. As QA has also shown promise for training BMs ([Adachi and Henderson, 2015](#); [Dixit et al., 2021](#)), our full model lends itself well for denoising images using quantum annealers, and could thus play a role in their future applications since QA can then be leveraged for *both* the training and denoising steps. The model also shows promise in absence of QA, and our insights presented are not limited to the QA framework, as the QUBO formulation of the denoising problem and its statistical properties we prove may be of independent interest.

The paper is organized as follows. Section 2 gives a summary of background on quantum annealing and Boltzmann Machines. Section 3 describes our main contribution of the image denoising model for QAs, and Section 4 shows some practical results obtained.

Remark 1.1. We frame our work as a binary image denoising method, although the framework does not depend on the data being images, and can be applied to the denoising of any binary data. This is because the framework does not use any spatial relationships between the pixels, and instead treats the image as a flattened vector whose distribution is to be learned. Hence, the denoising scheme can be applied as-is to any other binary data setting.

1.1. Contributions and organization

We provide QUBO-based denoising method for binary images (applicable to general binary data) using restricted Boltzmann machines in Section 3. This is done by formulating the denoising objective in equation 6 by combining the energy function of the distribution learned by the RBM with a (parameterized) penalty term for deviations from a given noisy image. This objective turns out to have an equivalent QUBO formulation, which is shown

in claim 1. In Theorem 3.4, we derive the optimal choice for the penalty parameter under the assumption that the true images follow the distribution learned by the RBM, which also recovers the maximum a posteriori estimate per Corollary 3.5, though our model is more flexible, and this flexibility allows for useful practical modifications. Theorem 3.6 shows that the denoising method yields a result that is *strictly* closer (in expectation) to the true image than the noisy image is, under some additional assumptions. Given that these idealistic assumptions won't be met in reality, we propose a robustness modification in Section 3.3 that *improves* performance empirically. In Section 4, as the method lends itself well to quantum annealing, we then implement the method on a D-Wave Advantage 5000-qubit quantum annealer, demonstrating strong empirical performance. Since only small datasets can be tested on the D-Wave machine due to the relatively low number of qubits, we also test the method on a larger dataset, for which we use simulated annealing on a conventional computer in place of quantum annealing to find good solutions the QUBO denoising objective. Though we highlight the method being well-suited for quantum annealers, we emphasize that it may be of independent interest to the machine learning and image processing communities at large.

1.2. Related work

Closely related work of [Koshka and Novotny \(2021\)](#) uses a similar model as ours for the image reconstruction task, also solving QUBO formulations via quantum annealing. In the reconstruction task, some subset of pixels is unknown (or obscured or missing), and needs to be restored, whereas our work considers denoising, where which pixels are noise-afflicted is unknown. [Greig et al. \(1989\)](#) derives a maximum a posteriori (MAP) estimator for the noise free image as a denoising method in a particular model of binary images that is less general than ours, though we would recover their estimator under a particular choice of our penalty parameter if we were to apply our framework to their model (since we recover MAP in a more general setting). Further, RBMs and quantum annealing have been studied for the classification problem, for instance in [Adachi and Henderson \(2015\)](#) and [Krzysztof et al. \(2021\)](#). Other research in the machine learning communities has also studied handling *label noise*, such as related work in [Vahdat \(2017\)](#), which studies the problem of training models in the presence of noisy labels, whereas our approach is entirely unsupervised (the data need not have any labels to begin with).

2. Background

Quantum Annealers make use of quantum interactions with the primary goal of finding the ground state of Hamiltonian by initializing and then evolving a system of coupled qubits over time ([Johnson et al., 2011](#)). In particular, we may view QA as implementing the Ising spin-glass model ([Nishimori, 2001](#)) evolving over time. As the QUBO model is equivalent to the Ising model ([Glover et al., 2018](#)), and QUBO instances can be efficiently transformed to Ising instances, a QA is well suited to provide good

solutions to QUBO problems. A QUBO cost function, or energy function, takes the form

$$f_Q(x) := \sum_{ij} Q_{ij}x_i x_j \tag{1}$$

where $x_i \in \{0, 1\}$, and Q is a symmetric, real-valued matrix. We will occasionally refer to Q_{ij} as the *weight* between x_i and x_j . QUBO is well-known to be NP hard (Barahona, 1982), and many combinatorial problems can be reformulated as QUBO instances. See Lucas (2014) and Glover et al. (2018) for thorough presentation of QUBO formulations of various problems. A Boltzmann Distribution using the above QUBO as its energy function takes the form

$$P_Q^{model}(x) = \frac{1}{z} \exp(-f_Q(x)), \tag{2}$$

where z is a normalizing constant. Note that a parameter called inverse temperature has been fixed to unity and is not explicitly shown in the above expression. In this paper, we will focus on making use of Boltzmann Machines, a type of generative neural network that fits a Boltzmann Distribution to the training data via making use of latent variables. Specifically, we consider Restricted Boltzmann Machines (RBMs), which have seen significant success and frequent use in deep probabilistic models (Goodfellow et al., 2016). RBMs consist of an input layer of *visible* nodes, and a layer of latent, or *hidden* nodes, which each have zero intra-group weights. Let $v \in \{0, 1\}^v$ and $h \in \{0, 1\}^h$ denote the visible and hidden nodes, respectively. It will be convenient for us to write $x = (v, h) \in \{0, 1\}^{v+h}$ as their concatenation. The probability distribution represented by a RBM is then

$$P_Q^{model}((v, h)) = \frac{1}{z} \exp(-f_Q(v, h)) \tag{3}$$

with the restriction that $Q_{ij} = Q_{ji} = 0$ if $i, j \in \{1, \dots, v\}$ or $i, j \in \{v + 1, \dots, v + h\}$. Hence, we have the simplified energy function

$$\begin{aligned} f((v, h), Q) &= \sum_{i=1}^{v+h} \sum_{j=1}^{v+h} 2Q_{ij}(v, h)_i (v, h)_j = \sum_{i=1}^v \sum_{j=v+1}^{v+h} Q_{ij}v_i h_j \\ &+ \sum_{i=1}^v Q_{ii}v_i^2 + \sum_{i=v+1}^{v+h} Q_{ii}h_i^2 \\ &= \mathbf{h}^T W \mathbf{v} + b_v^T \mathbf{v} + b_h^T \mathbf{h} =: f_{W, b_v, b_h}(v, h) \end{aligned} \tag{4}$$

where W is the $v \times h$ matrix consisting of the Q_{ij} weights between the visible and hidden nodes, and b_v and b_h are vectors of the diagonal entries Q_{ii} , $i \in \{1, \dots, v\}$ corresponding to visible nodes, and Q_{ii} , $i \in \{v + 1, \dots, v + h\}$ corresponding to hidden nodes, respectively. We will write the Boltzmann distribution with this energy function as P_{W, b_v, b_h} , noting that this is also P_Q^{model} for the appropriate Q .

It is well-known that RBMs can universally approximate discrete distributions (Goodfellow et al., 2016), making them a powerful model. They are also more easily trained than general Boltzmann Machines, usually through the contrastive divergence algorithm as described in Hinton (2002), or variants thereof.

2.1. Training Boltzmann Machines

We first devote some discussion to the training of RBMs. Subsection 3.1 then describes how to denoise images via QUBO given a well-trained RBM.

Continuing with the notation as in Equation (4), the probability distribution represented by a RBM is

$$P_\theta(v, h) = \frac{1}{z_\theta} \exp(-f_\theta).$$

For simplicity, denote $\theta = (W, b_v, b_h)$ as the model parameters henceforth. The normalizing constant z_θ above is

$$z_\theta = \sum_{v \in \{0,1\}^v} \sum_{h \in \{0,1\}^h} \exp(-f_\theta(v, h))$$

which becomes intractable quickly even for relatively small values of v and h . The common training approach aims to maximize the log-likelihood of the data. At a high-level, this will be done by approximating gradients and following a stochastic gradient scheme. However, since our data consists only of the visible nodes, we need to work with the marginal distribution of the visible nodes. This is given by

$$P_\theta(v) = \sum_h P_\theta(v, h) = \sum_h \frac{\exp[-f_\theta(v, h)]}{z_\theta}$$

Denote our set training data samples by $V := \{v^1, \dots, v^N\}$. We will use superscripts to indicate training data samples, and reserve subscripts to denote entries of vectors. Then the log-likelihood is given by

$$\begin{aligned} l_\theta(V) &= \sum_{k=1}^N \log P_\theta(v^k) = \sum_{k=1}^N \log \sum_h P_\theta(v^k, h) \\ &= \left(\sum_k \log \sum_h \exp(-f_\theta(v^k, h)) \right) - N \cdot \log z_\theta \\ &= \left(\sum_k \log \sum_h \exp(-f_\theta(v^k, h)) \right) \\ &\quad - N \cdot \log \sum_v \sum_h \exp(-f_\theta(v, h)) \end{aligned} \tag{5}$$

Now we can calculate the gradient with respect to θ as

$$\begin{aligned} \nabla l_\theta(V) &= \sum_{k=1}^N \frac{\sum_h \exp(-f_\theta(v^k, h)) \nabla(-f_\theta(v^k, h))}{\sum_h \exp(-f_\theta(v^k, h))} \\ &\quad - N \cdot \frac{\sum_{v, h} \exp(-f_\theta(v, h)) \nabla(-f_\theta(v, h))}{\sum_{v, h} \exp(-f_\theta(v, h))} \\ &= \sum_{k=1}^N \mathbb{E}_{P_\theta(h|v^k)} [-\nabla f_\theta(v^k, h)] - N \cdot \mathbb{E}_{P_\theta(v, h)} [-\nabla f_\theta(v, h)] \\ &= \frac{1}{N} \sum_{k=1}^N \mathbb{E}_{P_\theta(h|v^k)} [(v^k)^T \mathbf{h} + v^k + \mathbf{h}] \\ &\quad - \mathbb{E}_{P_\theta(v, h)} [v^T \mathbf{h} + v + \mathbf{h}] \end{aligned}$$

The first term can be computed exactly and efficiently from the data, since the conditional $P_\theta(h|v)$ admits the simple form $P(h_j = 1|v) = \text{logistic}(b_h + (v^T W)_j)$; we refer the interested reader to Goodfellow et al. (2016) or Dixit et al. (2021) and will focus on the second term. Due to its intractability to compute (one would have to sum over all possibilities of v and h), the most promising approach is to approximate it by sampling from $P_\theta(v, h)$. Classically, this is done via Gibbs sampling as described in Hinton (2002). However, recent research has also investigated using quantum annealers to sample from the relevant Boltzmann distribution, as suggested in Benedetti et al. (2015) and Dixit et al. (2021), which would make QAs useful in the training process since obtaining good Gibbs samples can be expensive. We note that together with our framework, QAs show promise to become useful for both the RBM training and the denoising process in the implementation of our method.

3. Image denoising as quadratic unconstrained binary optimization

This section is devoted to showing how one can naturally frame the image denoising problem as a QUBO instance over a learned Boltzmann Distribution fit to the data.

3.1. Denoising via QUBO

Let us assume we are given a trained Restricted Boltzmann Machine described in Section 2. The model prescribes to each vector $x \in \{0, 1\}^{v+h}$ the cost $f_Q(x)$ and corresponding likelihood $P_Q^{model}(x)$ defined in Equations (1) and (3), respectively. We will here make the assumption that P_Q^{model} describes the distribution of our data. Hence, high likelihood vectors in P_Q^{model} correspond to low cost vectors of f_Q . In particular, note that finding the maximum likelihood argument in Equation (2) corresponds to finding a solution to the QUBO instance in Equation (1).

Now, supposing this model, our goal is to reconstruct an image that has been affected by noise. The visible portion of our vector will be considered to be a flattened image with v pixels, black or white corresponding to 0 or 1, respectively, in the binary entries of the vector.

3.1.1. Noise model

We now describe the noise assumptions we will conduct our analysis under.

Definition 3.1. For $x \in \{0, 1\}^v$, we define x afflicted by salt-and-pepper noise of level σ as the random variable $\tilde{X}_{x,\sigma} := (x + \epsilon) \text{mod} 2$, where $\epsilon_i = B_i(p) \sim \text{Bern}(\sigma)$, independently.

In other words, a binary image afflicted by salt-and-pepper noise has each pixel independently flipped with probability σ . In particular, we are interested in $\tilde{X}_{X,\sigma}$, where $X \sim P_Q^{model}$, which is the compound random variable obtained by sampling X from the learned distribution of the data and then afflicting it with salt-and-pepper noise. For notational simplicity, will simply write \tilde{X} when the intended subscripts are clear from context.

We remark here that this salt-and-pepper noise model, also sometimes called impulse valued noise, is a natural choice for binary data and can occur in image processing through faulty sensors or pixel elements in cameras; see e.g., Boyat and Joshi (2015) for discussion of noise models in digital image processing. Since the pixels (or binary data entries for non-image binary data) only take the values 0 or 1, individual entries can only be corrupted by the value being flipped. Hence, continuous noise models such as Gaussian noise are not appropriate. Further, since the data we can work with on currently available quantum machines are very small, imposing additional structure on the noise does not seem fitting. However, the related problem of image reconstruction, in which some known set of pixels is damaged, is another model appropriate for such data, as studied in Koshka and Novotny (2021). We emphasize that in our noise model, which pixels are affected by noise is random and unknown, leading to the denoising problem.

Suppose we are given a realization $\tilde{x} \in \{0, 1\}^v$ of $\tilde{X}_{X,\sigma}$. The reconstruction process aims to retrieve this original X using \tilde{x} and the trained model through Q . The approach we will take begins from the intuition that X is likely to be a high-likelihood image that is close to \tilde{x} . To enforce this “closeness” to \tilde{x} while searching for higher likelihood images in our model to remove noise, we add to the cost in Equation (1) a penalty for deviations from \tilde{x} to formulate the following natural denoising cost function:

$$f_{Q,\tilde{x},\rho}(x) = f_Q(x) + \rho \sum_{ij} (x_i - \tilde{x}_i)^2 \tag{6}$$

for some $\rho > 0$ that determines the penalty level. The intuition is that the minimizer of this function for a well-chosen ρ will change a restricted number of pixels to find an image that is similar to the noisy image, but has a lower cost, i.e., higher likelihood, under the model, in hopes of removing the noise.

We show next that this minimizing Equation (6) corresponds to solving a QUBO instance.

Claim 1. Defining $\tilde{Q}^{\rho,\tilde{x}} \in \mathbb{R}^{(v+h) \times (v+h)}$ by setting $\tilde{Q}_{ij}^{\rho,\tilde{x}} = Q_{ij}$ if $i \neq j$ and $\tilde{Q}_{ij}^{\rho,\tilde{x}} = Q_{ii} + \rho(1 - 2\tilde{x}_i)$ if $i = j$, we have

$$\text{argmin}_x f_{Q,\tilde{x},\rho}(x) = \text{argmin}_x f_{\tilde{Q}^{\rho,\tilde{x}}}(x). \tag{7}$$

Proof.

$$\begin{aligned} f_{Q,\tilde{x},\rho}(x) &= f_Q(x) + \rho \sum_i (x_i - \tilde{x}_i)^2 = \sum_{ij} Q_{ij} x_i x_j \\ &+ \rho \sum_i x_i^2 - 2x_i \tilde{x}_i + \tilde{x}_i^2 \\ &= \sum_{i \neq j} Q_{ij} x_i x_j + \sum_i Q_{ii} x_i^2 + \rho(x_i^2 - 2x_i \tilde{x}_i + \tilde{x}_i^2) \\ &= \sum_{i \neq j} Q_{ij} x_i x_j + \sum_i (Q_{ii} + \rho(1 - 2\tilde{x}_i)) x_i^2 + \rho \tilde{x}_i^2 \\ &= f_{\tilde{Q}^{\rho,\tilde{x}}}(x) + \sum_i \rho \tilde{x}_i \end{aligned}$$

Noting that $x_i = x_i^2$ for the above derivation since they are in $\{0, 1\}$ here. Since the \tilde{x}_i terms do not depend on x , the claim follows.

Hence, solving the QUBO in on the right hand side of Equation (7) gives us the solution to Equation (6). Claim Equation 1 thus

tells us that we simply need to modify the diagonal of the original matrix Q of our model by adding $diag(1 - 2\tilde{x}_1, \dots, 1 - 2\tilde{x}_n)$ and then solve the resulting QUBO to get the denoised image. We can then make use of quantum annealing to solve the resulting QUBO of 7, or use classical methods and heuristics like simulated annealing instead. We formally spell out the denoising procedure in algorithm QUBO_Denoise.

QUBO_Denoise

Input: A matrix Q , a noisy image \tilde{x} sampled from the distribution of $\tilde{X}_{X,\sigma}$ with $X \sim P_Q^{model}$, and a penalty parameter $\rho > 0$.

Output: A denoised image $X_{\rho,\tilde{x},Q}^*$.

1. Set $\tilde{Q}_{ij}^{\rho,\tilde{x}} = Q_{ij}$ if $i \neq j$ and $\tilde{Q}_{ij}^{\rho,\tilde{x}} = Q_{ii} + \rho(1 - 2\tilde{x})$ if $i = j$.
2. Set $X_{\rho,\tilde{x},Q}^* = \text{argmin}_x f_{\tilde{Q}^{\rho,\tilde{x}}}(x)$.

For the remainder of the paper, $X_{\rho,\tilde{x},Q}^*$ will denote the denoised image obtained by applying QUBO_Denoise with noisy image \tilde{x} , penalty parameter ρ , and the distribution-defining matrix Q .

Remark 3.2. Considering the entire process of sampling a noisy image and then denoising it, the measurability of $X_{\rho,\tilde{x},Q}^*$ is inherited from the measurability of $\tilde{X}_{X,\sigma}$, which in turn inherits its measurability as compound random variable of the measurable noise and original image $X \sim P_Q^{model}$.

3.2. Optimal choice of penalty parameter ρ

The choice of the parameter ρ for the proposed image denoising model is clearly crucial to its success, since different choices will result in different solutions. If ρ is chosen to be too small, there is very little cost to flipping a pixel, and then many pixels may be flipped and the solution may not resemble the noisy image at all anymore. If ρ is too large, we may be too heavily penalizing flipping pixels, and thus may not be able to get rid of noise effectively. Hence, we now turn toward finding the optimal choice for ρ . We will evaluate the choice of ρ via *expected overlap*:

Definition 3.3. The *expected overlap* between two distributions P and a P' , is defined by

$$d(P, P') := \mathbb{E}_P \mathbb{E}_{P'} [n - \|X - X'\|_1],$$

where $X \sim P, X' \sim P'$.

We will consider $X \sim P_Q^{model}$, and X' as $X_{\rho,\tilde{x},Q}^*$ the corresponding denoised image, and will also call $d(P, P')$ the *expected overlap between X and X'* . To keep notation simple, for the remainder of this section allow us to write \tilde{X} in place of $\tilde{X}_{X,\sigma}$, with X and σ being clear from context.

Our main positive result concerning the choice of ρ is summarized in the following theorem:

Theorem 3.4. Let $X \sim P_Q^{model}$ as in Equation (2) and \tilde{X} be the noisy image. Then choosing $\rho = \log \frac{1-\sigma}{\sigma}$ to obtain $X_{\rho,\tilde{x},Q}^*$ is optimal with respect to maximizing the expected overlap between X and $X_{\rho,\tilde{x},Q}^*$.

Proof. Let $X \text{ dist } P_Q^{model}$, and \tilde{X} be X afflicted by salt-and-pepper noise of level σ . Then since $\tilde{X}_{X,\sigma}$ is obtained by flipping pixels with probability σ , we have the conditional probability

$$P_\sigma(\tilde{X} = \tilde{x} | X = x) = \prod_{i=1}^v \{ \sigma(\tilde{x}_i - x_i)^2 + (1 - \sigma)[1 - (\tilde{x}_i - x_i)^2] \} = \frac{\exp[-\beta_\sigma \sum_{i=1}^v (\tilde{x}_i - x_i)^2]}{(1 + e^{-\beta_\sigma})^v}, \tag{8}$$

where $\beta_\sigma := \log \frac{1-\sigma}{\sigma}$. In order to infer the original image X from the noisy one \tilde{X} , we utilize the Bayes formula and calculate the conditional probability $P_{\beta_\sigma, Q}^{post}(X = x | \tilde{X} = \tilde{x})$.

$$P_{\beta_\sigma, Q}^{post}(x | \tilde{x}) = \frac{P_\sigma(\tilde{X} = \tilde{x} | X = x) P_Q^{model}(x)}{\sum_{\{x\}} P_\sigma(\tilde{x} | x) P_Q^{model}(x)} = \frac{\exp[-\beta_\sigma \sum_{i=1}^v (\tilde{x}_i - x_i)^2 - \sum_{i,j=1}^{v+h} Q_{ij} x_i x_j]}{\sum_{\{x\}} \exp[-\beta_\sigma \sum_{i=1}^v (\tilde{x}_i - x_i)^2 - \sum_{i,j=1}^{v+h} Q_{ij} x_i x_j]}. \tag{9}$$

Note that x includes pixels for hidden nodes, which is fine here. Our approach finds the state which is most likely under this distribution, which is realized by annealing for the above QUBO with the β_σ term.

The overlap of two vectors x^* and x is given by

$$m(x, x^*) := \frac{1}{v+h} \sum_{i=1}^{v+h} (2x_i - 1)(2x_i^* - 1), \tag{10}$$

the proportion of shared entries. We consider the average (over the noise) of solutions, $\tilde{X}_{\rho,\tilde{x},Q}$ with

$$(\tilde{X}_{\rho,\tilde{x},Q})_i = \theta \left(\sum_{\{x\}} P_Q^{model}(x) x_i - \frac{1}{2} \right), \tag{11}$$

where $\theta(x) = 1$ if $x > 0$, otherwise 0, noting that the right hand side represents the inferred pixel value based on the expectation from P_Q^{model} . We have formally distinguished $P_Q^{model}(x)$ from $P_{\rho, Q}^{post}(x | \tilde{x})$, but in fact they are the same. Note that

$$2(\tilde{X}_{\rho,\tilde{x},Q})_i - 1 = \text{sign} \left(\sum_{\{x\}} P_Q^{model}(x) (2x_i - 1) \right), \tag{12}$$

where $\text{sign}(x)$ is the sign of x . Let $\alpha_{\sigma, Q} := -\beta_\sigma \sum_i (\tilde{x}_i - x_i)^2 - \sum_{i,j} Q_{ij} x_i x_j$ for conciseness. In order to evaluate the statistical performance of our method with coefficient ρ of penalty term, we calculate the average of overlap as

$$M_{\beta_\sigma, Q}(\rho) := \sum_{\{\tilde{x}\}, \{x\}} P_\sigma(\tilde{x} | x) P_Q^{model}(x) m(\tilde{X}_{\rho,\tilde{x},Q}, x) = \frac{1}{(1 + e^{\beta_\sigma})^v} \frac{1}{z} \frac{1}{v+h} \sum_i \sum_{\{\tilde{x}\}, \{x\}} e^{\alpha_{\sigma, Q}} [2(\tilde{X}_{\rho,\tilde{x},Q})_i - 1] (2x_i - 1). \tag{13}$$

A sum in the right hand side of the above equation holds

$$\begin{aligned}
 & \sum_{\{x\}} e^{\alpha_{\sigma,Q}} [2(\mathbb{E}(X_{\rho,\tilde{x},Q}^*) - 1)(2x_i - 1)] \\
 & \leq \left| \sum_{\{x\}} e^{\alpha_{\sigma,Q}} [2(\mathbb{E}(X_{\rho,\tilde{x},Q}^*) - 1)(2x_i - 1)] \right| \\
 & \leq \left| \sum_{\{x\}} e^{\alpha_{\sigma,Q}} (2x_i - 1) \right| = \sum_{\{x\}} e^{\alpha_{\sigma,Q}} (2x_i - 1) \\
 & \frac{\sum_{\{x\}} e^{-\beta_{\sigma} \sum_i (\tilde{x}_i - x_i)^2 - \sum_{ij} Q_{ij} x_i x_j} (2x_i' - 1)}{\sum_{\{x\}} e^{-\beta_{\sigma} \sum_i (\tilde{x}_i - x_i)^2 - \sum_{ij} Q_{ij} x_i x_j} (2x_i' - 1)} \\
 & = \sum_{\{x\}} e^{\alpha_{\sigma,Q}} (2x_i - 1) \text{sign} \left(\sum_{\{x'\}} P_Q^{\text{model}}(x') (2x_i' - 1) \right) \\
 & = \sum_{\{x\}} e^{\alpha_{\sigma,Q}} [2(\tilde{X}_{\rho,\tilde{x},Q})_i - 1](2x_i - 1).
 \end{aligned} \tag{14}$$

Hence, the averaged overlap holds

$$\begin{aligned}
 M_{\beta_{\sigma,Q}}(\rho) & \leq \frac{1}{(1 + e^{\beta_{\sigma}})^v} \frac{1}{Z_{1,Q}} \frac{1}{v + h} \sum_i \sum_{\{\tilde{x}\}, \{x\}} \\
 & e^{-\beta_{\sigma} \sum_i (\tilde{x}_i - x_i)^2 - \sum_{ij} Q_{ij} x_i x_j} [2(\tilde{X}_{\rho,\tilde{x},Q})_i - 1](2x_i - 1) \\
 & = M_{\beta_{\sigma,Q}}(\beta_{\sigma}).
 \end{aligned} \tag{15}$$

This inequality means that the averaged overlap is maximized when $\rho = \beta_{\sigma} = \log \frac{1-\sigma}{\sigma}$.

This theorem is based on a known fact in statistical physics of information processing (Nishimori, 2001) and translates the fact into the setting of our problem. Notably, the optimal choice of ρ does not depend on the distribution of the data, but only on the noise level, for which in many real world cases one may have good estimates. The proof of the theorem also reveals the following corollary:

Corollary 3.5. Under the same assumptions of Theorem 3.4, setting $\rho := \log \frac{1-\sigma}{\sigma}$ makes $X_{\rho,\tilde{x},Q}^*$ the maximum a posteriori estimator for the original noise-free image X .

The corollary follows from observing that the energy function in the numerator of the posterior distribution Equation (9) is exactly Equation (6) with $\rho := \frac{1-\sigma}{\sigma}$, noting that minimizing Equation (6) is equivalent to maximizing Equation (9). However, this framework allows for additional flexibility in choosing the ρ parameter that is absent in standard MAP estimation. In fact, in Sections 3.3 and 4.1 we go on to demonstrate that in practice, choosing a larger ρ may be beneficial for robustness of the method.

Though Theorem 3.4 derives the optimal choice of ρ , it does not give any guarantees that the method will yield an improvement in expected overlap, even under its assumptions. Next, we prove a theorem to show that in the case of visible units being independent of one another, our image denoising method produces in expectation strict denoising improvements with respect to the expected overlap. For $c > 0$ and a model distribution P_Q^{model} as in Equation 2, let \mathcal{I}_c be the set of indices i such that $|Q_{ii}| > c$.

These indices correspond to components of X that are either 0 or 1 with probability at least $\frac{1}{1 + e^{-c}}$, depending on whether Q_{ii} is positive or negative, respectively.

Theorem 3.6. Suppose that Q is diagonal, $X \sim P_Q$, and that \tilde{X} is X afflicted by salt-and-pepper noise of level σ . With \mathcal{I}_c as defined above for $c > 0$, setting $\rho \geq \log(\frac{1-\sigma}{\sigma})$, and assuming that $\mathcal{I}_{\rho} \neq \emptyset$, the expected overlap of the denoised image and the true image is strictly larger than the expected overlap of the noisy image and the true image, i.e.,

$$\mathbb{E} \left[\sum \mathbb{I}((X_{\rho,\tilde{x},Q}^*)_i = X_i) \right] > \mathbb{E} \left[\sum \mathbb{I}(\tilde{X}_i = X_i) \right]. \tag{16}$$

Proof. Let $\mathcal{I}_c^0 := \{i \in \mathcal{I}_c : Q_{ii} > 0\}$, $\mathcal{I}_c^1 := \{i \in \mathcal{I}_c : Q_{ii} < 0\}$. Intuitively, these are the indices which are likely to be zero or one, respectively. Further, letting $x^{\dagger i}$ denote the vector obtained by flipping entry i of x , we have that $|f_Q(x) - f_Q(x^{\dagger i})| = Q_{ii} > c$ if and only if $i \in \mathcal{I}_c$. Hence, this reveals that x^* solves Equation (6) by setting $x_i^* = 1 \ \forall i \in \mathcal{I}_{\rho}^1$, $x_i^* = 0 \ \forall i \in \mathcal{I}_{\rho}^0$, and $x_i^* = \tilde{x}_i$ otherwise, since the value of f_Q of Equation (1) is reduced by more than ρ , so that the overall penalized objective Equation (6) improves despite the ρ penalty accrued by the pixel flips.

Now, let $X \sim P_Q^{\text{model}}$. Let us compute $P((X_{\rho,\tilde{x},Q}^*)_i = X_i)$. The cases where this happens are: $i \in \mathcal{I}_{\rho}^0$ and $X_i = 0$, $i \in \mathcal{I}_{\rho}^1$ and $X_i = 1$, or $i \notin \mathcal{I}_{\rho}$ and pixel i was not flipped by the noise.

We know that if $i \in \mathcal{I}_{\rho}^b$, $P(X_i = b) \geq \frac{1}{1 + e^{-\rho}}$, for $b \in \{0, 1\}$, so $P((X_{\rho,\tilde{x},Q}^*)_i = X_i) \geq \frac{1}{1 + e^{-\rho}}$ for these. For $i \notin \mathcal{I}_{\rho}$, $P((X_{\rho,\tilde{x},Q}^*)_i = X_i) = 1 - \sigma$, where σ is the probability that the pixel was flipped by the noise. On the other hand, $P(\tilde{X}_i = X_i) = 1 - \sigma \ \forall i$. We characterize

$$\mathbb{E} \left[\sum \mathbb{I}((X_{\rho,\tilde{x},Q}^*)_i = X_i) \right] > \mathbb{E} \left[\sum \mathbb{I}(\tilde{X}_i = X_i) \right] \tag{17}$$

$$\sum P((X_{\rho,\tilde{x},Q}^*)_i = X_i) > \sum P(\tilde{X}_i = X_i) = n \cdot (1 - \sigma) \tag{18}$$

For the left-hand side, assuming $\mathcal{I}_{\rho} \neq \emptyset$, we have

$$\begin{aligned}
 \sum P((X_{\rho,\tilde{x},Q}^*)_i = X_i) & > \sum_{i \in \mathcal{I}_{\rho}} \frac{1}{1 + e^{-\rho}} + \sum_{i \notin \mathcal{I}_{\rho}} (1 - \sigma) \\
 & = |\mathcal{I}_{\rho}| \cdot \frac{1}{1 + e^{-\rho}} + (n - |\mathcal{I}_{\rho}|)(1 - \sigma)
 \end{aligned}$$

so that Equation (17) holds when

$$|\mathcal{I}_{\rho}| \cdot \frac{1}{1 + e^{-\rho}} + (n - |\mathcal{I}_{\rho}|)(1 - \sigma) \geq n(1 - \sigma)$$

$$\iff |\mathcal{I}_{\rho}| \neq 0$$

$$\text{and } \frac{1}{1 + e^{-\rho}} \geq 1 - \sigma \iff \rho \geq \log\left(\frac{1-\sigma}{\sigma}\right) \text{ and } \mathcal{I}_{\rho} \neq \emptyset, \tag{19}$$

and the theorem is proven.

The assumption that matrix Q is diagonal is equivalent to the components of X being independent, which is not realistic with real data. However, since in the RBM model the visible units are independent conditioned on the hidden units, we still consider

this independent case to be informative to the denoising method. In fact, if the hidden states were fixed (or known, or recovered correctly), Theorem 3.6 would apply. We leave it as a tantalizing open question to generalize this result beyond the independent case. The assumption of nonemptiness of \mathcal{I}_ρ is a natural one for the denoising task; indeed, when \mathcal{I}_ρ is empty, no entries of Q are large in magnitude, which is equivalent to the entries of X being close to uniformly distributed. In that case, intuitively of course it should not be possible to guarantee that we can denoise an image well if it looks like noise to begin with.

3.3. Robust choice of ρ

The optimal choice of ρ as derived in Theorem 3.4 relies on the assumption that the observed data comes from the learned distribution, or equivalently that the distribution generating our data has been perfectly learned by the RBM. However, in practice we will always only approximately learn the data distribution. Hence, we do not want to rely too heavily on the exact distribution we have learned when we denoise the images. One may hope to have a more robust method by only changing the value of a pixel when there is some confidence in the model that the pixel should be flipped. We may thus want to penalize flipping pixels slightly more than we should under the idealistic setting of Theorem 3.4, which corresponds to choosing a larger ρ value than $\log \frac{1-\sigma}{\sigma}$, or equivalently using a smaller $\sigma' < \sigma$ value when setting $\rho := \log \frac{1-\sigma'}{\sigma'}$. We opt for the latter as a means of intentionally biasing ρ to make the approach more robust for application. Figures 2, 3 in Section 4 show the effect this proposed robustness modification has, demonstrating indeed that choosing a larger ρ via intentionally using a smaller σ yields positive results. If the true noise level is σ , our experiments demonstrate that setting to roughly $\rho := \frac{1-0.75\sigma}{0.75\sigma}$ has a positive effect on performance.

4. Empirical results

This section contains results from implementing the previously described method and comparing it against other denoising approaches. Datasets and code are available on the first author's GitHub¹ for the purpose of easy reproducibility.

4.1. Datasets and setup

In this subsection, we present empirical results obtained by implementing our model on a quantum annealer, D-Wave's Advantage_system4.1, which has 5,000 qubits and enables embedding of a complete bipartite graph of size 172×172 . Hence, we use 12×12 pixel images here so that the visible layer is of size 144. We test the method on two different datasets with very differently structured data.

The first dataset is a 12×12 version of the well-known MNIST dataset (LeCun et al., 2010), created by downsizing the original dataset with nearest-neighbor image downscaling and binarizing

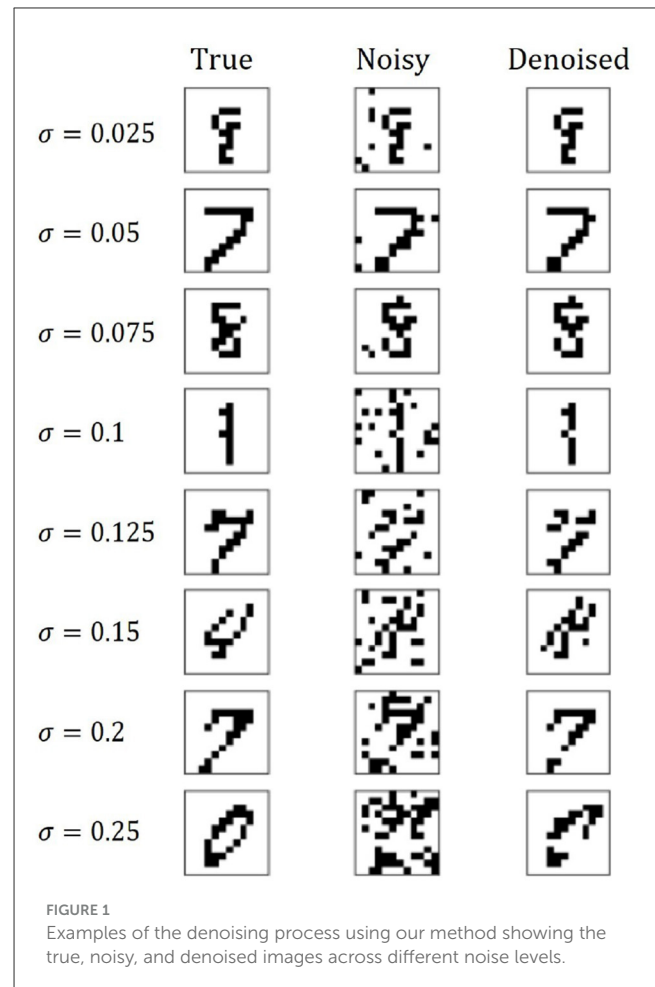


FIGURE 1 Examples of the denoising process using our method showing the true, noisy, and denoised images across different noise levels.

pixels. The second dataset we use is a 12×12 pixel Bars-and-Stripes (BAS) dataset, as has been used in closely related work (Dixit et al., 2021; Koshka and Novotny, 2021), in which the authors used a smaller 8×8 version of BAS in order to accommodate a 2,000 qubit machine, so we implement a larger 12×12 version for the 5,000 qubit machine we use. Each image consists of binary pixels with either each row or each column sharing the same values, so that each image consists of either “bars” or “stripes”. Some examples of noise-free, noisy, and denoised images across different noise levels are presented in Figure 1.

For both datasets we train the RBM by using the classical Contrastive Divergence algorithm first presented in Hinton (2002), and as described in Section 2.1. The number of hidden units was set to 50 and 64 for BAS and MNIST, respectively. For both datasets, we used learning rate of 0.01, batch size of 50, and 150 epochs as the training hyperparameters. For the BAS data, 4,000 images were generated as training data, and 1,000 as test data, while for MNIST, we simply used the full MNIST provided training set of 60,000 images and test set of 10,000 images. Noisy images were generated by adding salt-and-pepper noise of level σ to images from the test dataset. Given a noisy image, we are then able to embed and solve the resulting QUBO of 7 onto a D-Wave quantum annealer, Advantage_system4.1. A function of D-Wave's Ocean software, find_embedding, is utilized to find appropriate

¹ <https://github.com/PhillipKerger>

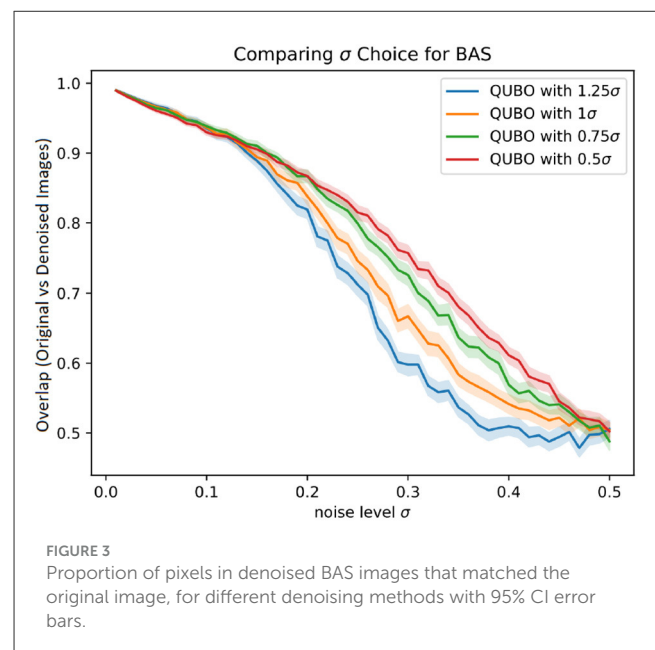
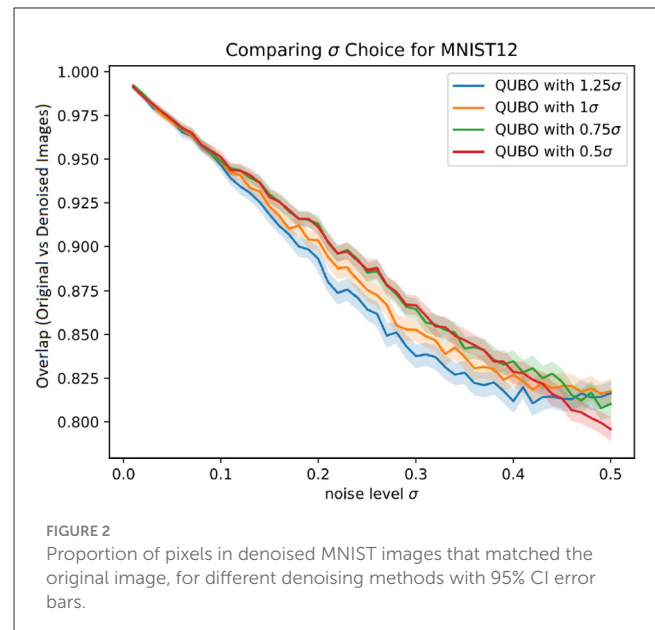
mappings from variables in a QUBO to physical qubits on D-Wave's Pegasus graph. A variable in QUBO is often mapped to multiple physical qubits, called chain, that are strongly connected to each other to behave like a single variable. A mapping can be used for every noisy images for each dataset, since their QUBO have the same graph structure. We have prepared in advance 50 sets of the different mappings for each dataset and choose a mapping from the pool at random to embed QUBO of each image. This random selection is done to avoid possible artificial effects on the denoising performance from using only a particular mapping. Parameters for embedding and annealing, i.e., `chain_strength` and `annealing_time`, are tuned to maximize the performance. In particular, we set `chain_strength` as the product of a coefficient c_0 and the maximum abstract value among the elements of each QUBO matrix, where we tune c_0 . The adopted values of the parameters are different between MNIST and BAS but the same values for all the range of σ . We set $(c_0, \text{annealing_time}) = (0.6, 50 \mu\text{s}), (0.5, 40 \mu\text{s})$ for BAS and MNIST, respectively. The number `num_reads` of reads of annealing is 100 for each noisy image. We calculate the average of solution of each pixel over the reads to approximate Equation (11) and use it to evaluate the overlap that is proportion of pixels in denoised images that matched the original image. We denoise 200 noisy images for each σ , which are randomly selected from the pool of test images for each sigma. Note also that for each value of sigma, the different methods compared use the same set of (randomly selected) noisy test images.

4.2. Results with quantum annealing

Figures 2, 3 first investigate the robust choice of ρ as discussed in Section 3.3. This is done by using a biased value of σ when setting $\rho = \log \frac{1-\sigma}{\sigma}$, instead setting $\rho := \log \frac{1-b\sigma}{b\sigma}$ for some bias factor b . The denoising performance for $b \in \{1.25, 1, 0.75, 0.5\}$ are shown, with 95% confidence intervals obtained by bootstrapping. Note that using a bias factor $b = 1$ means using the true value of σ for determining ρ .

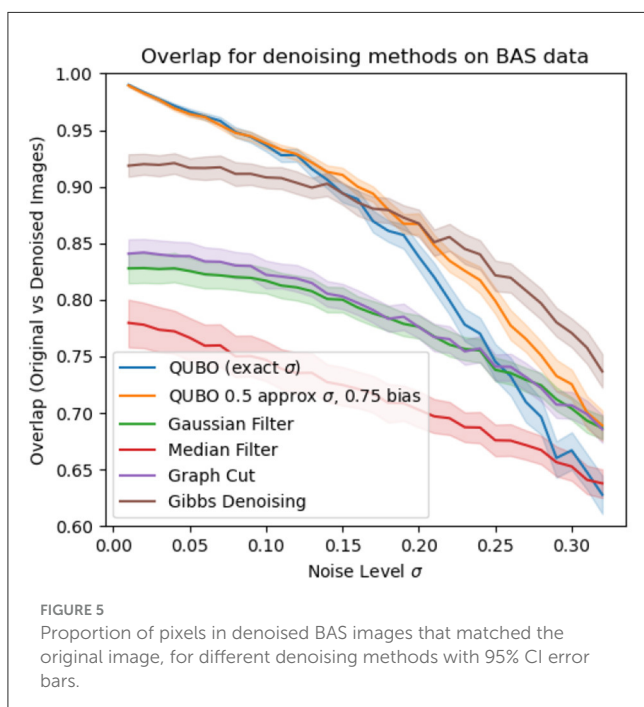
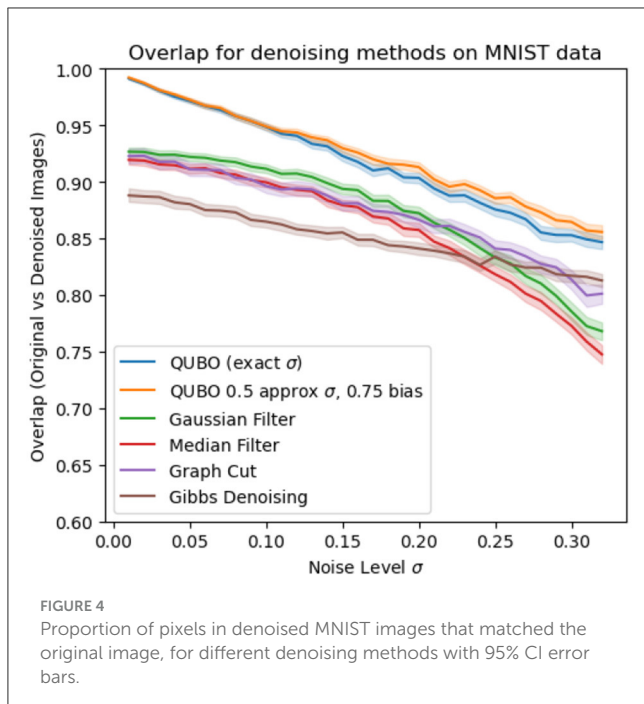
Based on the empirical performance, using a bias factor of around 0.75 seems to give an improved performance compared to using a bias factor of 1 in both data sets. A bias factor of 0.5 seems to perform quite well-across most noise regimes as well, with largely overlapping confidence regions to the 0.75 parameter setting, though in the low-noise setting for the BAS dataset we observe an adverse effect. The authors thus suggest a setting of 0.75 for the bias factor.

Next, in Figures 4, 5, we compare our method to popular other denoising methods for binary images on the 12×12 MNIST and bars-and-stripes datasets, respectively, across different noise levels. When comparing to other methods, a crucial factor is that we choose ρ based off of σ , but in practice σ may be unknown. In light of this, we include two versions of our method in these comparisons. First, we use our method with $\rho := \log \frac{1-\sigma}{\sigma}$, using the true value of σ without introducing the recommended bias factor. Secondly, we simulate the situation in which the true σ is unknown, and instead we only have a guess for σ . To simulate having an approximate guess for σ , for each image afflicted by noise of level σ , we sample σ' uniformly from an interval of size $\sigma/2$



centered at sigma. We then set $\rho := \log \frac{1-0.75\sigma'}{0.75\sigma'}$, using a bias factor of 0.75 on with this “guessed” value of σ . This is a significantly more realistic way of testing our method, since it gives an idea of how well the method may perform when the true noise level present in the noisy images is unknown and must be guessed. Our implementation here only assumes that the practitioner roughly knows the magnitude of the noise. For example, if the true noise is $\sigma = 0.2$, here we sample σ' uniformly from $[0.15, 0.25]$ to simulate the guess.

We compare our method to Gibbs denoising with an RBM (Tang et al., 2012, Section 3.2), median filtering (Huang et al., 1979), Gaussian filtering (Stockman and Shapiro, 2001, Chapter 5), and a graph-cut method (Greig et al., 1989) for denoising. For the Gibbs denoising, we use the same well-trained RBM as



for our QUBO-based method, and parameters of the method were carefully tuned for best performance to use 20 Gibbs iterations to then construct the denoised image as the exponentially weighted average of the samples with decay factor 0.8. Notably, as Gibbs-based denoising also requires a well-trained RBM, this method incurs the same computational overhead of training an RBM as our method does. However, it has the disadvantage of requiring careful tuning of the hyperparameters of the number of Gibbs iterations and decay factor to use, whereas our method of

picking ρ is much more straightforward and shows good results without tuning. For the graph-cut method, the recommended parameter setting in the reference of $\beta = 0.5$ is used. The median filter, Gaussian filter, and Gibbs denoising (excluding the overhead of training the RBM) each have complexity $O(n)$, where n is the number of pixels, whereas the graph-cut method has complexity $O(n^3)$ since a maximum-flow problem is solved on a graph whose nodes are the pixels of the image. Keeping the annealing time and number of reads as constant, the scaling of our method is also $O(n)$. We forego wall-time here, since the software implementations we compare against are specialized for large problems, so comparing walltime for the small problems that can be implemented on current quantum annealers may not be representative. However, we note that for the QUBO denoising as we use up to $50\mu s$ annealing time and 100 reads per image, denoising an image only takes a total of $5ms$ of annealing time in our case.

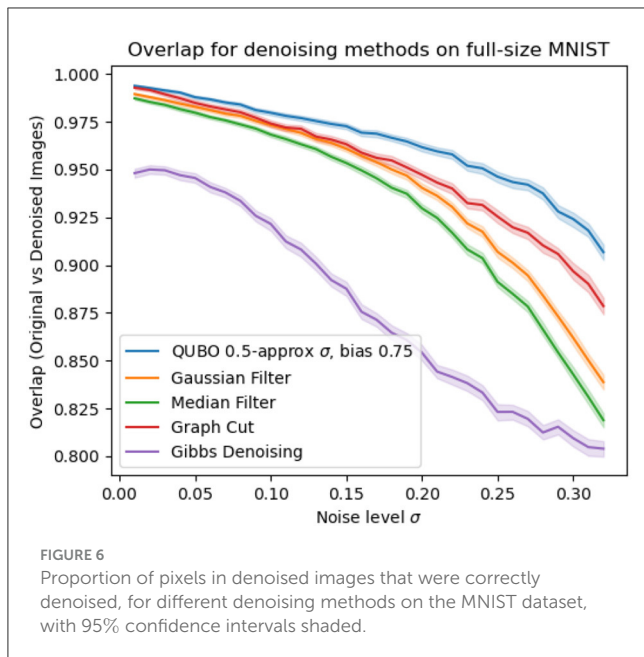
Results are summarized in Figures 4, 5. Overall, the QUBO-based method performs quite strongly. Across all noise regimes in the MNIST data, and in most noise regimes in the bars-and-stripes dataset, the method outperforms the others. In particular, for the MNIST data the 95% confidence region for the QUBO method entirely dominates the others. Indeed, we see the good performance that our analysis from Section 3 suggests, even when the true σ is unknown and instead guessed. Using a guessed σ and the robustness modification of Section 3.3 makes the method perform as well (if not slightly better) as knowing the true σ without the robustness modification. Only in the noise regime of $\sigma \geq 0.2$ in the BAS data does Gibbs denoising outperform our method.

4.3. Testing on larger images

Though we see the the straightforward implementability of our method on quantum annealers as a strong positive, a current drawback on using QAs is the limited data size that can be handled to accommodate their still small qubit capacities. Of course we can still instead test our method on larger datasets by obtaining solutions to the denoising QUBO 6 using other means. In Figure 6, we implement our method on a binarized version of the popular MNIST dataset (LeCun et al., 2010) by using simulated annealing (Kirkpatrick et al., 1983) to find solutions to (6). We particularly choose to test on the full-size MNIST dataset since we could only use a downscaled version on the QA due to size limitations on the input data, so this experiment serves to test our method without this downscaling. All methods are implemented as described in Section 4.1, and again for our method we use a guessed σ to simulate the unknown σ case and bias the guess for robustness.

5. Conclusion and future work

We investigated an image denoising framework via a penalty-based QUBO denoising objective that shows promise both theoretically through its statistical properties and practically through its empirical performance together with



the proposed robustness modification. The method is well-suited for implementability on a quantum annealer, providing an important application of QAs within machine learning through the fundamental image denoising task. Good results are still obtained on larger datasets when the QUBO is only classically approximated by simulated annealing instead, revealing the approach to be promising even in the absence of QAs. As RBMs form a core building block of many deep generative models such as deep Boltzmann machines or deep belief networks (Goodfellow et al., 2016), a natural next step is to attempt to incorporate this approach into these more complex models, though current hardware limitations on existing quantum annealers are restrictive. Further, since our method takes advantage of QAs for the denoising step, further research into making use of QAs for the training process of RBMs would yield a full image denoising model where both the model training and image denoising make use of QA.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://github.com/PhillipKerger/Code_for_figures_of_QUBO_RBM_denoising_paper.

References

- Adachi, S. H., and Henderson, M. P. (2015). Application of quantum annealing to training of deep neural networks. *ArXiv*. ArXiv:1510.06356. doi: 10.48550/arXiv.1510.06356
- Albash, T., and Lidar, D. A. (2018). Adiabatic quantum computing. *Rev. Mod. Phys.* 90, 15002. doi: 10.1103/RevModPhys.90.015002

Author contributions

PK: Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Software, Validation, Visualization, Writing—original draft, Writing—review and editing. RM: Conceptualization, Formal analysis, Funding acquisition, Project administration, Resources, Software, Supervision, Writing—review and editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. PK was supported in part by g-RIPS Sendai, Cyberscience Center at Tohoku Univ., and NEC Japan, in early stages of the work. PK is grateful to the USRA Feynman Academy internship program, support from the NASA Academic Mission Services (contract NNA16BD14C), and funding from DARPA under DARPA-NASA agreement SAA2-403688.

Acknowledgments

The early stage of this work is based on the work in the g-RIPS Sendai 2021 program. The authors thank Y. Araki, E. Escobar, T. Mihara, V. Q. H. Huynh, H. Kodani, A. T. Lin, M. Shirane, Y. Susa, and H. Suito for collaboration in the program. The authors also acknowledge H. Kobayashi and M. Sato for the use of the computing environment in the program. PK thanks Y. Sukurdeep for helpful feedback and discussions.

Conflict of interest

RM was employed by NEC Corporation.

The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- a case study with possible applications in deep learning. *Phys. Rev. A* 94, 022308. doi: 10.1103/PhysRevA.94.022308
- Boyat, A. K., and Joshi, B. K. (2015). A review paper: noise models in digital image processing. *ArXiv*. doi: 10.5121/sipij.2015.6206
- Buades, A., Coll, B., and Morel, J.-M. (2005). A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.* 4, 490–530. doi: 10.1137/040616024
- Cho, K. (2013). Boltzmann machines and denoising autoencoders for image denoising. *ArXiv*. ArXiv:1301.3468. doi: 10.48550/arXiv.1301.3468
- Das, A., and Chakrabarti, B. K. (2008). Colloquium: quantum annealing and analog quantum computation. *Rev. Mod. Phys.* 80, 1061–1081. doi: 10.1103/RevModPhys.80.1061
- Dixit, V., Selvarajan, R., Alam, M. A., Humble, T. S., and Kais, S. (2021). Training restricted boltzmann machines with a d-wave quantum annealer. *Front. Phys.* 9, 589626. doi: 10.3389/fphy.2021.589626
- Glover, F. W., Kochenberger, G. A., and Du, Y. (2018). Quantum bridge analytics i: a tutorial on formulating and using qubo models. *4OR* 17, 335–371. doi: 10.1007/s10288-019-00424-y
- Goodfellow, I. J., Bengio, Y., and Courville, A. (2016). *Deep Learning*. Cambridge, MA: MIT Press. Available online at: <http://www.deeplearningbook.org> (accessed August 20, 2023).
- Greig, D., Porteous, B., and Scheult, A. H. (1989). Exact maximum a posteriori estimation for binary images. *J. R. Stat. Soc. Ser. B Methodol.* 51, 271–279. doi: 10.1111/j.2517-6161.1989.tb01764.x
- Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Comput.* 14, 1771–1800. doi: 10.1162/089976602760128018
- Huang, T. S., Yang, G., and Tang, G. (1979). A fast two-dimensional median filtering algorithm. *IEEE Transact. Acoust. Speech Signal Process.* 27, 13–18. doi: 10.1109/TASSP.1979.1163188
- Johnson, M. W., Amin, M. H. S., Gildert, S., Lanting, T., Hamze, F., Dickson, N., et al. (2011). Quantum annealing with manufactured spins. *Nature* 473, 194–198. doi: 10.1038/nature10012
- Kadowaki, T., and Nishimori, H. (1998). Quantum annealing in the transverse ising model. *Phys. Rev. E* 58, 5355–5363. doi: 10.1103/PhysRevE.58.5355
- Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science* 220, 671–680. doi: 10.1126/science.220.4598.671
- Koshka, Y., and Novotny, M. A. (2021). Comparison of use of a 2000 qubit d-wave quantum annealer and mcmc for sampling, image reconstruction, and classification. *IEEE Transact. Emerg. Top. Comp. Intell.* 5, 119–129. doi: 10.1109/TETCI.2018.2871466
- Krzysztof, K., Mateusz, S., Marek, S., and Rafał, R. (2021). Applying a quantum annealing based restricted boltzmann machine for mnist handwritten digit classification. *Comp. Methods Sci. Technol.* 27, 99–107. doi: 10.12921/cmst.2021.0000011
- LeCun, Y., Cortes, C., and Burges, C. (2010). *Mnist Handwritten Digit Database*. ATT Labs. Available online at: <http://yann.lecun.com/exdb/mnist> (accessed August 20, 2023).
- Lucas, A. (2014). Ising formulations of many NP problems. *Front. Phys.* 2, 5. doi: 10.3389/fphy.2014.00005
- Nishimori, H. (2001). *Statistical Physics of Spin Glasses and Information Processing: An Introduction*. New York, NY: Oxford University Press.
- Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D* 60, 259–268. doi: 10.1016/0167-2789(92)90242-F
- Stockman, G., and Shapiro, L. G. (2001). *Computer Vision*. Upper Saddle River, NJ: Prentice Hall PTR.
- Tang, Y., Salakhutdinov, R., and Hinton, G. E. (2012). “Robust boltzmann machines for recognition and denoising,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition* (Providence, RI), 2264–2271.
- Vahdat, A. (2017). “Toward robustness against label noise in training deep discriminative neural networks,” in *NeurIPS Proceedings 2017* (Long Beach, CA).