HYBRID QUANTUM-CLASSICAL STOCHASTIC NETWORKS WITH BOLTZMANN LAYERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Quantum Machine Learning (QML) has the potential to significantly advance the state-of-the-art in artificial intelligence, due to recent developments in quantum computing hardware and algorithm design. Particularly, an avenue opened up by these advances is the possibility of enhancing classical models through developing quantum analogues, which have greater representational power at no extra cost in terms of training and inference. Here, we investigate analogues of classical networks with stochastic layers, by introducing a class of hybrid stochastic networks that combine layers of several types, including stochastic quantum and classical layers and deterministic classical layers. Further, we introduce Quantum-Annealing (QA)-based sampling techniques that allow such models to be efficiently learned on current QA architectures, using variational and importance-sampling based approaches. Our framework provides benefits in training existing models, including Quantum Boltzmann Machines (QBMs) and Quantum Variational Autoencoders, by allowing local transverse field weights to be optimized jointly with other model parameters, and allows novel hierarchical hybrid models to be learned efficiently. We use classical simulations on synthetic and genomics data to test the impact of including quantum mechanical transverse field terms in such models relative to their classical counterparts. We show that hybrid models are able to achieve better predictive accuracy compared to classical models of matching architecture in these settings, and provide evidence that the local transverse terms can be interpreted as introducing tunable higher-order interactions by connecting genes belonging to common biological pathways.

1 INTRODUCTION

Recent approaches to quantum machine learning (QML) have leveraged the potential of quantum circuits and models to efficiently represent complex distributions and functions to enhance classical machine-learning approaches, including traditional regression-based approaches [2], Support Vector Machines [6], and Deep Neural-Networks [1,3,5,6,9]. The latter are examples of Quantum Deep Learning approaches, which include both gate-based algorithms [3,5] and Quantum-Annealing-based methods [1,9]. Further, both gate- and annealing-based approaches make use of hybrid classical-quantum methods during training. For the former, methods have been developed based on the Variational Quantum Eigensolver [8] to train deep quantum circuits [3,5,6], while the latter use a quantum analogue of the variational evidence lower-bound (Q-ELBO) [1,9].

Classical deep neural networks may be deterministic or stochastic, where stochastic networks are those that include probabilistic latent variables at internal layers and/or non-deterministic weights [15,18,19]. Quantum (neural) networks may also evolve deterministically or stochastically, with the former being primarily associated with gate-based circuit network models, and the latter associated with annealing-based approaches that use thermalization to sample from a desired quantum distribution over observables. However, the possibility of using quantum distributions to generate latent variables in a stochastic network is rarely considered, with the exception of the Quantum Variational Autoencoder (QVAE) [9], which includes a single layer of latent variables distributed according to a Quantum Boltzmann Machine (QBM) [1]. Here, we consider the case of stochastic networks containing multiple layers of probabilistic latent variables, whose layer-wise joint distributions may be derived from a classical or quantum models or sub-networks, and which may also include classical deterministic layers. We call such models 'Hybrid Stochastic Networks' (HSNs). Such models have

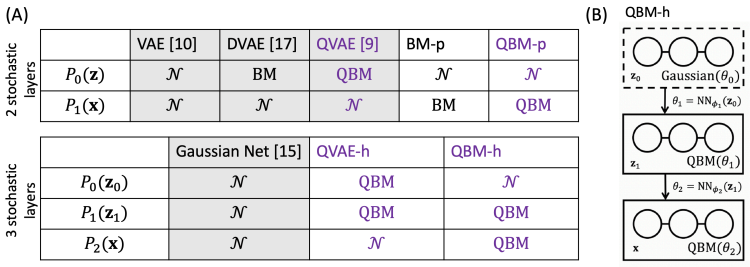

Figure 1: Summary of Hybrid Stochastic Networks explored in paper, and existing models. (A) shows combinations of stochastic layers used for various different HSN models ($\mathcal{N}$: Normal, BM: Boltzmann machine, QBM: Quantum Boltzmann machine layers, -p: structured parameter space, -h: hierarchical, magenta: hybrid quantum-classical, grey: existing model). (B) shows detailed schematic for the hierarchical-QBM (QBM-h) model. Notation as in Eq. 1.

intrinsic interest, since using probabilistic latent variables potentially enhances interpretability of the latent representations of the network, while maintaining the enhanced flexibility of quantum models. However, while techniques for training arbitrarily deep circuit-based QNN models have been developed [3,5], as noted, for hybrid-stochastic quantum networks the focus has been on architectures with a single stochastic quantum layer, such as the QVAE. Further, the Q-ELBO bound used during QVAE training does not permit direct optimization of the transverse weights in the embedded QBM, since the gradient of the bounded approximation always drives the transverse weights to zero [1].

In response to the above, we introduce a general framework for learning hybrid stochastic quantum-classical networks of arbitrary depth. Our framework naturally embeds previous models (QBMs and QVAEs) as well as allowing more powerful models to be built with multiple latent stochastic layers. Particularly, we focus on models which contain various combinations of classical Gaussian and QBM layers, and we define and investigate Hierarchical-QBMs and QBMs with structured parameter spaces (QBM-h and QBM-p, respectively). Further, we introduce an efficient representation of the clamped phase of the QBM; this allows clamped statistics (including those of the transverse weights) to be evaluated directly by drawing samples from a quantum annealer, while requiring only $O(\log(T))$ space, where T is the implicit number of Trotter slices underlying the representation. As we show, this representation may be used in the context of score-function gradient [13] and reparameterization gradient-based methods [9,10,17] for training hybrid stochastic models, including joint optimization of the transverse weights.

We demonstrate our method's efficiency and representational accuracy using classical simulations on synthetic data. Particularly, we show that increasing model expressivity allows models in our framework to capture structure more effectively than shallow models, and how the accuracy of our clamped QBM representation compares to continuous time QMC approaches for sampling. We further test the ability of multiple models in our framework to extract structure from psychiatric genomics data, showing that our hybrid approach learns generalizable structure more effectively than entirely classical models on matched architectures. Further, we consider the interpretability of the quantum models trained on genomics data; particularly, we provide evidence that the local transverse terms allow tunable higher-order interactions to be learned by connecting genes into common biological pathways.

2 Hybrid Quantum-Classical Stochastic Networks

We first define the Hybrid Quantum-Classical Stochastic Network model in general form. We formulate it as a generative model in which the inputs are unobserved latent variables, although these may be treated as observed in a supervised setting. For a model with L levels of latent variables:

$$\begin{aligned} \mathbf{z}_0 &\sim P_0(.;\theta_0) \\ \mathbf{z}_1 &\sim P_1(.;\theta_1 = \mathrm{NN}_{\phi_1}(\mathbf{z}_0)) \\ &\dots \\ \mathbf{x} &\sim P_L(.;\theta_L = \mathrm{NN}_{\phi_L}(\mathbf{z}_{L-1})). \end{aligned} \tag{1}$$

Here, $\text{NN}_\phi(.)$ denotes a classical neural network (of arbitrary width and height) with parameters ϕ, and $\mathbf{z}_{0...L-1}$, $\mathbf{x}$ are vector-valued random variables, which may be continuous or discrete. The model is fully specified by the set of distributions $P_0...P_L$, which may be classical or quantum (containing at least one quantum layer), and parameters $\theta_0, \phi_1..\phi_L$. We focus particularly on the case where each P_l is either a classical Gaussian, or a Quantum Boltzmann Machine (QBM). For a Gaussian P_l, we have $\theta_l = \{\mu_l, \mathbf{\Sigma}_l\}$, where $\mathbf{\Sigma}_l$ may be a symmetric, diagonal or full covariance matrix. For a QBM P_l (over a binary observed/latent vector), $\theta_l = \{\mathbf{b}_l, \mathbf{W}_l, \gamma_l\}$, where $\mathbf{b}$ and γ are real vectors of length N, where N is the number of qubits in the layer, and $\mathbf{W}$ is a symmetric matrix of real values with zero diagonal (dropping the layer suffixes for convenience). Given this parameterization, the QBM is associated with the following quantum Hamiltonian:

$$H(\theta) \quad = \quad \sum_i \gamma_i \sigma_i^{(x)} - \sum_i b_i \sigma_i^{(z)} - \sum_{i,j} W_{i,j} \sigma_i^{(z)} \sigma_j^{(z)}, \tag{2}$$

where i, j range over the qubits $1...N$, and we use the notation:

$$\sigma_a^{(z)} \quad = \quad \overbrace{I \otimes ... \otimes I}^{a-1 \text{ times}} \otimes \sigma_z \otimes \overbrace{I \otimes ... \otimes I}^{N-a \text{ times}}, \tag{3}$$

where $I = [1\ 0\ ;\ 0\ 1]$, $\sigma_z = [1\ 0\ ;\ 0\ -1]$, and $\sigma_a^{(x)}$ is defined as in Eq. 3, where $\sigma_x = [0\ 1\ ;\ 1\ 0]$ is substituted for σ_z (here, σ_z and σ_x are Pauli matrices). The parameters $\mathbf{b}$ and $\mathbf{W}$ specify the local field biases and couplings of the qubits in the *computational basis*, while γ specifies the strength of a transverse field local to each qubit (setting $\gamma = 0$ results in a classical Boltzmann Machine, BM). The probabilistic model for the QBM is specified via the density matrix, $\rho(\theta) = Z^{-1} \exp(-H(\theta))$, where Z is the partition function, $Z = \text{Tr}[\exp(-H(\theta))]$. The probability of a joint configuration $\mathbf{x}$ (or $\mathbf{z}_l$ if $l < L$) corresponds to the probability that measuring the qubits in the computational basis (the z-basis) will result in the configuration $\mathbf{x}$ being observed. This corresponds to a partial trace:

$$P_{\text{QBM}}(\mathbf{x}; \theta) \quad = \quad \text{Tr}[\Lambda_{\mathbf{x}} \rho(\theta)], \tag{4}$$

where $\Lambda_{\mathbf{x}}$ limits the trace to only those configurations consistent with $\mathbf{x}$ (which will be a single configuration if $\mathbf{x}$ has length N). Note that in this definition of a QBM, the spin observables take on the values $\{-1, 1\}$, but due to the prevalence of the $\{0, 1\}$ convention in the machine learning literature, all the following methods are presented with spins taking on values $\{0, 1\}$. The transformation between the two conventions is easily achieved.

We briefly note some special forms of the model in Eq. 1 (see Fig. 1A). If all distributions are Gaussian, we have a Deep Latent Gaussian Model, as defined in [15], which for the case $L = 1$ reduces to a traditional VAE [10]. If $L = 1$ and we set P_0 to a classical Boltzmann Machine, and P_1 is Gaussian, we recover a Discrete-VAE [17], while if P_0 is a QBM and P_1 Gaussian, we have a Quantum-VAE [9]. We may also define a hybrid model with $L = 1$, where P_0 is Gaussian and P_1 is a QBM, which we call a *QBM with structured parameter space* (QBM-p). Notice that while a QVAE models continuous observations, a QBM-p is a model of discrete (binary) observations. Further, we also define a hybrid *Hierarchical QBM* (QBM-h), with $L > 1$, setting P_0 to be Gaussian, and all other distributions to QBMs (which models discrete observations, see Fig. 1B), and a *Hierarchical QVAE* (QVAE-h) with $L > 1$, setting P_L to be Gaussian, and all others to QBMs (which models continuous observations).

Below we discuss training for models in the class defined by Eq. 2. A common requirement for a number methods is the need to evaluate gradients of the parameters of a QBM conditioned on a given observed output. In Sec. 2.2, we introduce an efficient representation for approximating such gradients via sampling; for convenience we discuss this method in the context of a generic Monte-Carlo estimator of the gradient of the ELBO and Q-ELBO bounds on the log-likelihood discussed in Sec. 2.1. Sec. 2.3 then discusses how our sampling approach may be applied in the context of score-function and reparameterization gradient-based methods.

2.1 Log-likelihood Bounds

Evidence Lower Bound (ELBO). We briefly review here the evidence lower bound (ELBO), which is typically optimized when training classical latent variable models [10,15,17,19]. For the model in

Eq. 2, we may write the ELBO as:

$$\begin{aligned}\mathcal{L}_{\text{ELBO}} &= \mathbb{E}_{Q(\mathbf{z}|\mathbf{x})}[\log P(\mathbf{x},\mathbf{z})] + H(Q(\mathbf{z}|\mathbf{x})), \\ &= \mathbb{E}_{Q(\mathbf{z}_{L-1}|\mathbf{x})}[\log P_L(\mathbf{x}|\mathbf{z}_{L-1})] + \sum_{l=1}^{L-1} \mathbb{E}_{Q(\mathbf{z}_l,\mathbf{z}_{l-1}|\mathbf{x})}[\log P_l(\mathbf{z}_l|\mathbf{z}_{l-1})] + \\ &\quad \mathbb{E}_{Q(\mathbf{z}_0|\mathbf{x})}[\log P_0(\mathbf{z}_0)] - \mathbb{H}(Q(\mathbf{z}|\mathbf{x})), \end{aligned} \tag{5}$$

where $\mathbb{H}(.)$ is the Shannon entropy, and $Q(\mathbf{z}|\mathbf{x})$ is a variational distribution over the latent variables, which typically has a simple form. For a layer l with a QBM distribution, Eq. 4 may be substituted into the term involving P_l in Eq. 5. We can then express the gradient with respect to the parameters as:

$$\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial \theta_l} = \mathbb{E}_{Q(\mathbf{z}_l,\mathbf{z}_{l-1}|\mathbf{x})}\left[\frac{\partial \log(\text{Tr}[\Lambda_{\mathbf{x}} \exp(-H(\theta_l))])}{\partial \theta_l} - \frac{\partial \log(\text{Tr}[\exp(-H(\theta_l))])}{\partial \theta_l}\right]. \tag{6}$$

A generic Monte-Carlo estimator for Eq. 6 is formed by taking the difference of the partial derivatives of the two terms on the RHS, each estimated by averaging over samples from $Q(\mathbf{z}_l, \mathbf{z}_{l-1}|\mathbf{x})$. The two terms are analogues of the 'clamped' and 'unclamped' distribution used in classical Boltzmann machine training. For the QBM, the unclamped term may be estimated as in the classical model by using samples from $P_{\text{QBM}}(.;\theta_l)$ to estimate the required partial derivatives for each sample from $Q(\mathbf{z}_l, \mathbf{z}_{l-1}|\mathbf{x})$. However, the partial derivatives for the clamped term cannot be estimated analogously to the classical case (see [1,9]); for small systems, the gradients of the log-trace may be explicitly calculated, although we discuss alternative sampling strategies in Sec. 2.2.

Quantum Evidence Lower Bound (Q-ELBO). To allow gradients for the QBM to be calculated analogously to the classical BM, [1,9] introduce a Quantum analogue to the ELBO, which we state in general form below for the HSN model. This is based on the Golden-Thompson inequality: $\text{Tr}[e^A e^B] \geq \text{Tr}[e^{A+B}]$ for matrices A and B. Applying this to the generative QBM log-likelihood:

$$\tilde{P}_l(\mathbf{z}_l;\theta) = \frac{\text{Tr}[\exp(-H_{\mathbf{z}_l}(\theta))]}{\text{Tr}[\exp(-H(\theta))]} \leq P_l(\mathbf{z}_l;\theta). \tag{7}$$

where $H_{\mathbf{z}}(\theta) = H(\theta) - \log(\Lambda_{\mathbf{z}})$ is the Hamiltonian of a 'clamped' QBM. The lower-bound $\tilde{P}_l$ in Eq. 7 may be substituted for the terms involving QBM distributions P_l in the ELBO bound, resulting in a lower-bound on the ELBO (and hence also the log-likelihood), $\mathcal{L}_{\text{Q-ELBO}}$. As shown in [1,9], this substitution allows the partial derivatives for the first term in Eq. 6 to be estimated by using samples from the clamped-QBM based on the Hamiltonian $H_{\mathbf{z_l}}(\theta)$. However, while this allows the b_i and w_{ij} QBM parameters to be trained, the gradient for the γ_i's is always negative, meaning that the transverse weights cannot be trained using this bound since they will always be driven to zero (hence in [1], a global transverse field term is optimized as a hyperparameter).

2.2 World-line Representations

As discussed, a Monte-Carlo estimate for the Q-ELBO gradient may be derived by sampling from QBMs with the unclamped and clamped Hamiltonians, $H(\theta)$ and $H_{\mathbf{x}}(\theta)$, respectively (for convenience, we consider a QBM over the output layer, $\mathbf{x}$ in the following). Conveniently, these Hamiltonians may be represented using pairwise energy models over N qubits, allowing samples to be drawn efficiently by Quantum Annealing, using for instance a D-Wave Annealer [1]. Both Hamiltonians may also be simulated classically. Most directly, the QBM can be approximated by a classical Hamiltonian formed by expanding the system over T 'Trotter slices'; for instance, the unclamped model corresponds to the following classical BM:

$$\begin{aligned} E(\mathbf{x}_{0\ldots T-1};\theta) &= \sum_{i,k} \gamma'_i [x_{i,k} = x_{i,k+1}] - \sum_{i,k} b'_i x_{i,k} - \sum_{i,j,k} w'_{i,j} [x_{i,k} = x_{j,k}], \\ P_{\text{QBM}}(\mathbf{x};\theta) &\approx \frac{\exp(-E(\mathbf{x};\theta))}{\sum_{\mathbf{x}} \exp(-E(\mathbf{x};\theta))} \end{aligned} \tag{8}$$

where k ranges over the $0...T-1$ (with periodic boundary conditions, slice $T \equiv$ slice 0), $\gamma'_i \approx \log(\gamma_i/T)/2$, $b'_i = 2b_i/T$, $w'_{i,j} = 2w_{i,j}/T$, and $[.]$ is the Iverson bracket, which is 1 for a true proposition, and 0 otherwise. The clamped model may be simulated similarly by clamping the visible

units in all Trotter slices to their observed values, and the approximation becomes exact as $T \to \infty$. The continuous limit may also be simulated classically using Continuous Time Quantum Monte Carlo (CT-QMC) [16] and Population Annealing (PA) [9, 11]. We run a simplified version of the CT-QMC+PA approach for the unclamped case to evaluate classical, annealing-based simulation methods for calculating the gradient statistics to train quantum models. For the PA analysis, our simplification is to use the variable temperature parameter only in the replica reweighting process, without scaling the parameters at each step (as was done in [9]). We demonstrate that CT-QMC+PA is able to recover the ground-truth gradient statistics in a synthetic model for a range of γ values, to an accuracy of $< 3e-4$ in mean-squared error (App. A.1, Fig. 3C).

While sampling from $H_{\mathbf{x}}(\theta)$ allows the Q-ELBO gradient to be estimated, classical simulation may also be used to estimate the partial derivatives of the term $\log(\text{Tr}[\Lambda_{\mathbf{x}} \exp(-H(\theta))])$ in the ELBO gradient (Eq. 6). This can be achieved by evaluating the expected statistics needed for the partial derivatives relative to the distribution below (following [9], and assuming all N units are visible), which we refer to as 'partially-clamped':

$$\begin{aligned} E(\mathbf{x}_{1\ldots T-1}|\mathbf{x}_0;\theta) &= \sum_{i,k} \gamma'_i[x_{i,k} = x_{i,k+1}] - \sum_{i,k>0} b'_i x_{i,k} - \sum_{i,j,k>0} w'_{i,j}[x_{i,k} = x_{j,k}], \\ \text{Tr}[\Lambda_{\mathbf{x_0}} \exp(-H(\theta))] &\approx \frac{\exp(-E(\mathbf{x}_{1\ldots T-1}|\mathbf{x}_0;\theta))}{\sum_{\mathbf{x}_{1\ldots T-1}} \exp(-E(\mathbf{x}_{1\ldots T-1}|\mathbf{x}_0;\theta))} \end{aligned} \tag{9}$$

Since Eq. 9 is a classical pairwise BM, samples from it may also be drawn using a physical Quantum Annealer by letting $\gamma \to 0$ for all qubits. This may be advantageous for small systems, since it avoids the need for classical Gibbs sampling (or other Monte-Carlo approaches), and leverages any potential benefits of tunneling in QA [4]. However, the need to replicate the system T times (requiring space $O(NT)$), means that for large N and T the number of logical qubits required will exceed the capacity of current physical annealers ($\sim$2000 physical qubits on D-Wave's 2000Q system). For this reason, below (and in App. A.1) we introduce a compressed discrete representation for the partially-clamped QBM, requiring only space $O(N(\log(T) + D))$ (for a degree D graph underlying W).

Auxiliary energy for restricted world-lines. We may approximate Eq. 9, by representing the model as an energy over world-lines, $[x_{i,1}, ..., x_{i,T-1}]$. For small γ_i, we may assume that the number of flips between 0 and 1 along each world-line are small, and hence a given world-line may be efficiently represented by a set of break-points denoting the Trotter slices at which the flips occur, which must be even in number to enforce cycle consistency. This representation is used in classical MCMC simulations for unclamped distributions (see [21]). In the clamped case, we assume that at most two break-points occur per world-line, denoted (u_i, v_i), where $u_i, v_i \in \{0...T-1\}$, $u_i \leq v_i$, and $u_i = t$ implies a 0/1 flip occurs between slices t and $t+1$ (similarly for v_i) unless $u_i = v_i$. Necessarily, x_0 is fixed to its observed value. We show in App. A.1 that this energy over world-lines can be represented as a binarized pairwise energy, by using $B = \log(T)$ bits to represent u_i and v_i each, and introducing one auxiliary binary variable per world-line and two per pair of world-lines with a non-zero w_{ij} coupling. The compressed representation has a classical energy of the form:

$$E(\mathbf{u}, \mathbf{v}, \mathbf{a}) = -\sum_{i,\alpha} \psi_i(u_i^\alpha, v_i^\alpha, a_i) - \sum_{(i,j)\in G,\alpha} \psi_{i,j}(u_i^\alpha, u_j^\alpha, v_i^\alpha, v_j^\alpha, a_{i,j}^{(1\ldots4)}), \tag{10}$$

where α ranges from $1...B$, $u_i = \sum_\alpha 2^{B-\alpha} u_i^\alpha$ (similarly for v_i), G is the set of edges with non-zero w_{ij} couplings, and the potential functions ψ_i and $\psi_{i,j}$ are each boolean polynomials of degree two. Since each world-line requires only $2N\log(T)$ bits, and up to 4 auxiliary bits are required per pairwise edge ($|G| = ND/2$ if each node is connected to D others), the number of qubits required to represent Eq. 10 is $O(N(\log(T) + D))$. As noted in Appendix A.1, not all configurations of $\mathbf{u}, \mathbf{v}, \mathbf{a}$ will result in valid world-line configurations in the partially-clamped QBM model. For this reason, rejection sampling must be used for non-valid configurations. However, we show that the representation may be fine-tuned to trade off the number of samples rejected versus the variance of the Monte-Carlo gradient estimator by an importance-sampling based approach. Particularly, we show that, by tuning the acceptance probability, the auxiliary energy can efficiently estimate the necessary gradient statistics for QBM training with an accuracy comparable to CT-QMC+PA (App. A.1, Fig 3B-C; note that the relative efficiency of drawing samples using CT-QMC+PA and our auxiliary energy implemented on a physical quantum annealer is likely to depend strongly on system size). Further, Appendix A.1 shows that the auxiliary energy may be expanded to include an arbitrary number of breakpoints, at the cost of an increased space complexity.

2.3 ALGORITHMS

Above, we have discussed the application of QBM sampling approaches to generate generic Monte-Carlo estimates for the ELBO and Q-ELBO gradients in Sec. 2.1. We now briefly consider alternative methods to which the sampling-based approaches of Sec. 2.2 may be applied. Most directly, variance reduction methods may be applied, such as score function gradients, to the Monte-Carlo estimates outlined [13,14]. Further, the reparametrization trick for discrete latent variables introduced in [17], and used in the context of QVAE training in [9], may also be used to learn HSNs having a QBM only as P_0 and all other layers Gaussian. Models of this form may be optimized using the Q-ELBO by directly extending the methods of [9]. In this context, however, our efficient auxiliary restricted world-line energy representation may also be applied to avoid lower-bounding the QBM terms in the ELBO objective, allowing efficient training of local transverse terms which is not permitted by the Q-ELBO bound. Finally, our sampling methods may also be applied to train HSNs using recent importance sampling and multi-sampling approaches [12,18], which have not previously been explored in the context of QML models. In the experimentation (Sec. 3) we focus on score-function gradient methods, which are used for both quantum and classical models (see Appendix A.2 for details).

3 RESULTS

3.1 DENSITY ESTIMATION

Synthetic Data. We begin by comparing HSN models with different architectures and distributions on a synthetic density estimation task. Here, we particularly investigate (a) the relative performance of quantum and classical models with matching architectures, (b) the effects of adding hierarchical structure, and (c) the performance of quantum models trained using the auxiliary restricted world-line energy from Sec. 2.2. For this purpose, we use the following synthetic task: The training, validation and test data each consist of 100 data points, each being a binary vector of length 8. The binary vectors are generated by (a) choosing uniformly from 20 prototype vectors (each generated by uniform sampling), and (b) adding 'correlated noise' to the prototype. To generate the correlated noise, 5 pairs of bits are predefined by uniform sampling (common to all prototypes), and with probability 1/3, each pair is activated for a given data point. For those pairs activated, the prototype bits are flipped at those positions (for intersecting pairs, the common bits are only flipped once). We then test the ability of each model to represent the underlying distribution, which by design incorporates a complex web of dependencies between pairs of bits and the prototype patterns. We compare the performance of 4 HSN models on this data: (1) a BM-p with a latent space of dimension 2 ($\mathbf{z}_0$), a fully connected output BM over 8 bits ($\mathbf{x}$), and a classical NN with two hidden layers of 20 units connecting the two; (2) a QBM-p with the same architecture as (1), along with individual γ terms for each of the output units; (3) a QBM-h, which adds an intermediate latent stochastic QBM layer ($\mathbf{z}_1$) over 4 qubits, and connects each pair of stochastic layers with two level, 20 unit, classical NNs; and (4) a model identical to (3), but trained with the auxiliary energy from Sec. 2.2, labeled QBM-h-aux. We optimize all models using score-function gradients and Gibbs sampling, and monitor the validation error as an early stopping criterion to stop training. The log-likelihood is estimated on the test-set using an Approximate Bayesian Computation estimator (see Appendix A.2 for details), and 5 Trotter slices are used to simulate all quantum models. The results for all models over 5 synthetic datasets are shown in Fig. 2A. As shown, there is a clear separation in performance between the hierarchical (3-level) and non-hierarchical (2-level) models ($p < 0.001$, ANOVA, and $p < 0.05$, paired t-test). The QBM-p performance is slightly higher than the BM-p, and the QBM-h-aux is marginally lower than the QBM-h ($p < 0.05$ and n.s. respectively, paired t-test); in the latter case, this suggests that the auxiliary energy function does not have a significant adverse affect on training the QBM-h model.

Genomics Data. We duplicate the set-up used above to analyze psychiatric genomics data from the PsychENCODE project [20], consisting of gene expression (RNA-Seq) levels from post-mortem prefrontal cortex samples of control, schizophrenia (SCZ), bipolar (BDP) and autistic (ASD) subjects. We select restricted subsets of the data by choosing $N = 50$ or $N = 200$ samples each for training, validation and testing sets, balanced for control and SCZ subjects, and selecting 8 genes out of the 'high-confidence schizophrenia genes' subset found in [20], which are most strongly correlated with SCZ (*SLC35G1, SERPINA3, GFAP, SLC14A1, C2CD2, CP, C4A, PENK*). To replicate the set-up for the synthetic task defined above, we binarize each gene's expression by thresholding it at the median

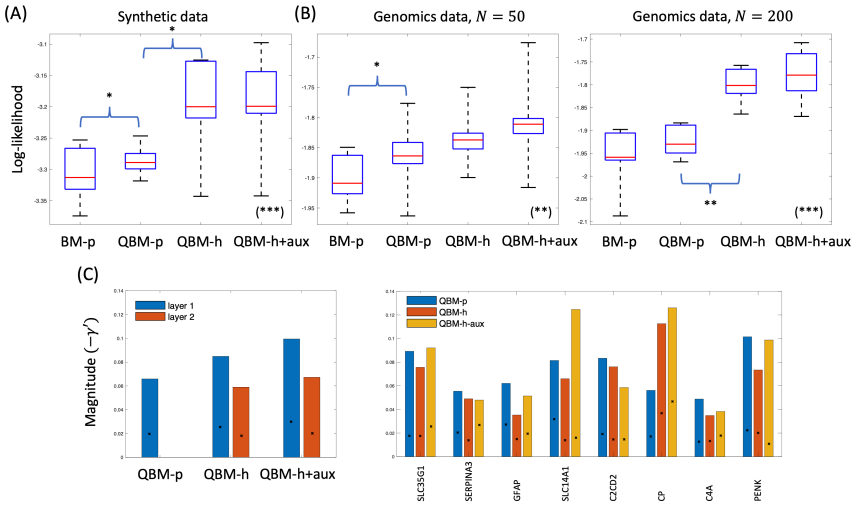

Figure 2: Results for density estimation, and model interpretation of γ-terms. (A) shows test-set likelihood of models on synthetic data, and (B) on psychiatric genomics data density estimation tasks. Bracketed significance levels are shown for 1-Factor ANOVA tests across models, while braces show pairwise comparisons (1-tailed paired t-test). Box plots are evaluated over 5 data partitions, with median, inter-quartile and 0.9-quantile ranges shown. (C) shows distribution of $-\gamma'$-terms across models/layers and grouped by gene. Bars show 0.75 percentile magnitudes, and crosses show median values. See text for discussion.

Table 1: Comparing classification accuracy on psychiatric genomics data. Table shows mean and standard deviation for the classification of case-control status using a kernel classifier in the latent space of each of the model classes (single), and a classifier based on a weighted combination of kernels in the latent and observed spaces (joint).

Kernel	BM-p	QBM-p	QBM-h	QBM-h-aux
Single	0.645 ± 0.027	0.65 ± 0.024	0.648 ± 0.028	0.65 ± 0.031
Joint	0.667 ± 0.032	0.667 ± 0.035	0.668 ± 0.036	0.67 ± 0.037

value across all subjects. Fig. 2B shows the corresponding model performances on the genomics data where differing amounts of data are used for testing and training, across 5 partitions of the data. As shown, each exhibits a similar pattern of model performances relative to the synthetic data. Additionally, the gain in performance of the QBM-p relative to the BM-p model appears to be more evident in the small dataset size ($N = 50$), suggesting that the extra expressive power and associated biases of the QBM-p model (including implicit higher-order interactions, as explored below) is useful for identifying generalizable structure in the small data setting. Further, the separation between hierarchical and non-hierarchical models is accentuated in the large-data setting (QBM-p vs. QBM-h, $p < 0.01$, 2-sample t-test), indicating that these models are able to more fully exploit their increased capacity in this setting. Finally, we also compare the ability of quantum and hierarchical HSN models to model continuous gene expression data densities using the VAE, QVAE and QVAE-h network models (see Fig. 1), defined using identical architectures to the BM, QBM-p and QBM-h models above, with binary and continuous units exchanged where relevant. These achieve estimated log-likelihoods of -297/-293, -260/-279, and -239/-263 for training/testing using VAE, QVAE and QVAE-h models respectively (mean across 5 folds), showing that our hierarchical framework can also improve the fit of models with continuous outputs, such as the QVAE [9].

3.2 Classification of Psychiatric Disorders

We further investigate each model by testing the ability of the representations learned in the latent space to perform classification of case-control status (for Schizophrenia) using the genomics data above ($N = 200$). We use a kernel classifier, calculating the Mahalanobis distance between a

given test point and all training points using the $\mathbf{z_0}$ returned for each data point by the encoder, and predicting the class by taking the weighted mean across the vectors $[0\ 1]$ and $[1\ 0]$, representing cases and controls respectively, while optimizing the variance of the Mahalanobis kernel σ on the validation partition. The results are shown in Table. 1 in the 'single' kernel line (note that, due to the data balancing, chance is 0.5). As shown, the quantum and hierarchical models achieve slightly better predictive performance, although the increase is smaller here than in the case of density estimation. Further, we compared to logistic regression and kernel predictors using the original raw features $\mathbf{x_0}$. These give performances of 0.645 ± 0.27 and 0.665 ± 0.029 respectively. Notably, while the logistic predictor is lower than the models above, the kernel predictor on the raw features is slightly higher; we thus tested whether the latent space kernels of the trained models are capturing complementary information beyond the kernel on the raw features by optimizing a joint kernel consisting of a weighted combination of the raw-features and latent-space kernels for each model on the validation partition. The 'joint' kernel line in Table. 1 shows the results for these combinations, showing that indeed the joint kernel is able to enhance performance, although the improvements are again small.

3.3 Interpretation of γ-terms

Given the enhanced performance of the hybrid quantum-classical models compared to classical models with matching architectures, we were interested to investigate possible interpretations of the the novel model parameters introduced in the hybrid networks, namely the γ_i transverse terms. We thus investigated the statistics of these parameters across models, both plotting their characteristic magnitudes per layer in each hybrid model, and per gene in the output stochastic layer across models (see Fig. 2C). We note that, in order to estimate these distributions, we feed-forward the $\mathbf{z}_0$ latent vectors encoding each training instance, and collect the $-\gamma_i' \approx -\log(\gamma_i/T)/2$ parameters for the QBM distributions generated by the decoder (the couplings between the Trotter slices). As shown, the hierarchical models exhibit a characteristic pattern whereby the $-\gamma'$ terms in the first stochastic layer are generally larger than those in the output layer. Further, the per-gene $-\gamma_i'$ parameters show a remarkable consistency across models, with the genes *SLC35G1, SLC14A1, CP, PENK* having notably higher values across models than the others. Particularly, this group includes two members of the *SLC* family of membrane transport proteins, and the membranous/extra-cellular synaptic protein *PENK*. Potentially, increased $-\gamma_i'$ terms for these genes permits the model to use implicit higher-order interactions between them, by strengthening the connections between the Trotter slices on the world-lines for these genes, while other genes rely primarily on intra-slice pairwise interactions. We perform an investigation of a simple synthetic energy in Appendix A.3 to demonstrate how the introduction of transverse terms can distort the classical pairwise energy to generate an effect similar to the introduction of a higher-order (non-pairwise) potential into the classical energy. Potentially, therefore, large $-\gamma_i'$ magnitudes (couplings) may be used to identify genes engaged in higher-order interactions (epistasis) involving biological pathways significant for a disorder.

4 Discussion

We have introduced a framework for constructing hybrid stochastic networks, with layers of probabilistic latent variables governed by both classical and quantum distributions. Further, we have introduced methods for learning such networks, which allow the transverse γ terms in all layers to be optimized jointly using the ELBO bound, by formulating an efficient representation for the partially-clamped QBM distribution; a number of existing models fall into our framework (such as the QBM and QVAE), which may be optimized using our approach. Future directions include adaptation of our model for testing on a physical QA architecture (including an additional mapping from logical qubits to physical qubits [2]). Further, additional types of quantum layer may be included in the model, defined not only by QBMs, but more generally by a quantum circuit, using a gate-based model; in this case, the techniques we have developed may be ported to optimize gate-based models with latent variables, which are currently under-explored [3]. Finally, we plan to explore further the interpretability of hybrid networks trained using our approach. Particularly, we have suggested that the γ terms may consistently distort the energy to allow higher-order interactions to be learned between features; we plan to investigate this phenomenon further across domains, as well as the potential for using domain-specific knowledge [20] to place priors on the latent structure of both the pairwise and transverse terms based on these observations.

REFERENCES

[1] Amin, M.H., Andriyash, E., Rolfe, J., Kulchytskyy, B. and Melko, R., 2018. Quantum boltzmann machine. Physical Review X, 8(2), p.021050.

[2] Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N. and Lloyd, S., 2017. Quantum machine learning. Nature, 549(7671), pp.195-202.

[3] Broughton, M., Verdon, G., McCourt, T., Martinez, A.J., Yoo, J.H., Isakov, S.V., Massey, P., Niu, M.Y., Halavati, R., Peters, E. and Leib, M., 2020. Tensorflow quantum: A software framework for quantum machine learning. arXiv preprint arXiv:2003.02989.

[4] Denchev, V.S., Boixo, S., Isakov, S.V., Ding, N., Babbush, R., Smelyanskiy, V., Martinis, J. and Neven, H., 2016. What is the computational value of finite-range tunneling?. Physical Review X, 6(3), p.031015.

[5] Farhi, E. and Neven, H., 2018. Classification with quantum neural networks on near term processors. arXiv preprint arXiv:1802.06002.

[6] Havlíček, V., Córcoles, A.D., Temme, K., Harrow, A.W., Kandala, A., Chow, J.M. and Gambetta, J.M., 2019. Supervised learning with quantum-enhanced feature spaces. Nature, 567(7747), pp.209-212.

[7] Johansson, J. R., Nation,P.D. and Nori, F., 2013. QuTiP 2: A Python framework for the dynamics of open quantum systems. *Comp. Phys. Comm.*, 184, p. 1234.

[8] Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J.M. and Gambetta, J.M., 2017. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. Nature, 549(7671), pp.242-246.

[9] Khoshaman, A., Vinci, W., Denis, B., Andriyash, E., Sadeghi, H. and Amin, M.H., 2018. Quantum variational autoencoder. *Quantum Science and Technology*, 4(1), p.014001.

[10] Kingma, D.P., Mohamed, S., Rezende, D.J. and Welling, M., 2014. Semi-supervised learning with deep generative models. In Advances in neural information processing systems (pp. 3581-3589).

[11] Machta, J. 2010. Population Annealing with Weighted Averages: A Monte Carlo Method for Rough Free-Energy Landscapes. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics* 82 (2 Pt 2), p. 026704.

[12] Mnih, A. and Rezende, D.J., 2016. Variational inference for monte carlo objectives. arXiv preprint arXiv:1602.06725.

[13] Paisley, J., Blei, D. and Jordan, M., 2012. Variational Bayesian inference with stochastic search. arXiv preprint arXiv:1206.6430.

[14]Ranganath, R., Gerrish, S. and Blei, D.M., 2014. Black Box Variational Inference. In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics.

[15] Rezende, D.J., Mohamed, S. and Wierstra, D., 2014. Stochastic backpropagation and approximate inference in deep generative models. arXiv preprint arXiv:1401.4082.

[16] Rieger, H., and N. Kawashima, 1999. Application of a Continuous Time Cluster Algorithm to the Two-Dimensional Random Quantum Ising Ferromagnet. *The European Physical Journal B - Condensed Matter and Complex Systems* 9 (2), p. 233–36.

[17] Rolfe, J.T., 2016. Discrete variational autoencoders. arXiv preprint arXiv:1609.02200.

[18] Tang, C. and Salakhutdinov, R.R., 2013. Learning stochastic feedforward neural networks. In *Advances in Neural Information Processing Systems*, pp. 530-538.

[19] Tran, D., Hoffman, M.D., Saurous, R.A., Brevdo, E., Murphy, K. and Blei, D.M., 2017. Deep probabilistic programming. *ICLR*, 2017.

[20] Wang, D., Liu, S., Warrell, J., Won, H., Shi, X., Navarro, F. C., Clarke, D., Gu, M., Emani, P., ... & Gerstein, M. (2018). Comprehensive functional genomic resource and integrative model for the human brain. *Science*, 362(6420), eaat8464.

[21] Wessel, S., 2013. Monte Carlo Simulations of Quantum Spin Models. In *Emergent Phenomena in Correlated Matter*, Online lecture notes at: https://www.cond-mat.de/events/correl13/manuscripts/wessel.pdf.

[22] Mnih, A. and Gregor, K., 2014. Neural variational inference and learning in belief networks. arXiv preprint arXiv:1402.0030.

[23] Tavaré, S., Balding, D.J., Griffiths, R.C. and Donnelly, P., 1997. Inferring coalescence times from DNA sequence data. Genetics, 145(2), pp.505-518.

A APPENDIX

A.1 AUXILIARY ENERGY FOR RESTRICTED WORLD-LINES

We show here how the 'partially-clamped' energy in Eq. 9 can be expressed in the form of Eq. 10. To do so, we consider an energy across the variables $u_{i=1\ldots N}$ and $v_{i=1\ldots N}$, where $0 \leq u_i < v_i < T$ or $u_i = v_i = 0$ (where N and T are the number of qubits and Trotter slices respectively; note that we index the qubits from $1...N$ and the Trotter slices from $0...T-1$). For convenience, we will also write $\mathbf{u}_i = (u_i, v_i)$, and $\mathbf{u} = \{\mathbf{u}_{i=1\ldots N}\}$. Here, u_i and v_i represent break-points in each column of clamped energy Trotter expansion, $E(x_{1\ldots T-1}|x_0)$. Hence:

$$\begin{aligned} x_{i,1\ldots u_i} &= x_{i,0}, \\ x_{i,u_i+1\ldots v_i} &= 1 - x_{i,0}, \\ x_{i,v_i+1\ldots T-1} &= x_{i,0}. \end{aligned} \tag{11}$$

We can write the energy of a given configuration implied by this representation directly as an energy over $\mathbf{u}$:

$$E_2(\mathbf{u}) = -\sum_i \phi_i(\mathbf{u}_i) - \sum_{i,j} \phi_{i,j}(\mathbf{u}_i, \mathbf{u}_j). \tag{12}$$

Here, ϕ_i and $\phi_{i,j}$ are respectively unary and pairwise potentials across columns in the original Trotter formulation. These have the form:

$$\begin{aligned} \phi_i(\mathbf{u}_i) &= -2\gamma_i'[v_i > u_i] + b_i^u(v_i - u_i), \\ \phi_{i,j}(\mathbf{u}_i, \mathbf{u}_j) &= w_{i,j}^{(1)}|\bar{\mathbf{u}}_i \cap \bar{\mathbf{u}}_j| + w_{i,j}^{(2)}|\bar{\mathbf{u}}_i \cap \mathbf{u}_j| + w_{i,j}^{(2)}|\mathbf{u}_i \cap \bar{\mathbf{u}}_j| + w_{i,j}^{(1)}|\mathbf{u}_i \cap \mathbf{u}_j| \end{aligned} \tag{13}$$

We use the set notation here to operate on the columns $\mathbf{u}_i$, so that as a set, $\mathbf{u}_i$ represents the (discrete) interval $\{u_i + 1 \ldots v_i\}$, and $\bar{\mathbf{u}}_i$ represents its complement, $\{1 \ldots T-1\} \backslash \mathbf{u}_i$ (see Fig. 3A for schematic). The coefficients in Eq. 13 are defined as:

$$\begin{aligned} b_i^u &= b_i'(1 - 2x_{i,0}), \\ w_{i,j}^{(1)} &= w_{i,j}'[x_{i,0} = x_{j,0}], \\ w_{i,j}^{(2)} &= w_{i,j}'[x_{i,0} \neq x_{j,0}]. \end{aligned} \tag{14}$$

This gives us a reformulation of the energy of Eq. 9, since $E_2(\mathbf{u}) = E(\mathbf{x}(\mathbf{u})|\mathbf{x}_0) + C$, writing $\mathbf{x}(\mathbf{u})$ for the configuration of $\mathbf{x}_{1\ldots T-1}$ corresponding to $\mathbf{u}$.

We wish to represent Eq. 14 in a binarized form (corresponding to Eq. 10). To do so, we introduce binary variables to represent $\mathbf{u}$, $u_i^{\alpha=1\ldots B}$ and $v_i^{\alpha=1\ldots B}$, where u_i^α is the α most significant bit in the binary representation of u_i. Hence, $B = \log_2 T$, and $u_i = \sum_\alpha 2^{B-\alpha} u_i^\alpha$. Further, we introduce auxiliary binary variables $\mathbf{a}$, which will be used as part of the binary representation, and implicitly allow each energy potential in the model to have local copies of the u_i^α and v_i^α variables (to be explicitly defined below). We now specify the binarized auxiliary energy function, E_3, as introduced in the main paper (Eq. 10), where we use $\mathbf{u}_{\text{bin}}$, $\mathbf{v}_{\text{bin}}$ and $\mathbf{a}$ to collectively refer to sets of binarized variables, and G to refer to the graph of pairwise edges:

$$E_3(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a}) = -\sum_{i,\alpha} \psi_i(u_i^\alpha, v_i^\alpha, a_i) - \sum_{(i,j)\in G,\alpha} \psi_{i,j}(u_i^\alpha, u_j^\alpha, v_i^\alpha, v_j^\alpha, a_{i,j}^{(1\ldots4)}), \tag{15}$$

We now define the explicit forms of the binary potentials, ψ_i and $\psi_{i,j}$. For ψ_i, we set:

$$\psi_i(u_i^\alpha, v_i^\alpha, a_i) = a_i \cdot 2^{B-\alpha}(b_i^u + c_i)(v_i^\alpha - u_i^\alpha) + (1 - a_i)(2\gamma_i' - K(u_i^\alpha + v_i^\alpha)). \tag{16}$$

Here, we introduce an auxiliary binary variable a_i, which represents, for a given column, whether $v_i > u_i$. If so, then the potential represents the cost or benefit incurred by the unary terms between the breakpoints, otherwise it represents the benefit gained by having no breakpoints in the column ($2\gamma_i'$). Further, we also introduce a large constant K; this is used to drive both the column breakpoints towards zero if $a_i = 0$, hence enforcing $v_i = u_i$. We note that the constraint, $v_i \geq u_i$ will only be satisfied stochastically. If $b_i^u > 0$, this constraint will tend to be satisfied, since the energy will be lower in configurations for which $v_i \geq u_i$. However, when $b_i^u < 0$ the reverse is true. For this reason, we allow the positive constant c_i to be added to the unary cost in Eq. 16, whose value can be optimized to allow efficient importance sampling as discussed below.

$$\begin{aligned} \psi_{i,j}(u_i^\alpha, u_j^\alpha, v_i^\alpha, v_j^\alpha, a_{i,j}^{(1\ldots4)}) = {} & (w_{i,j}^{(2)} - w_{i,j}^{(1)}) 2^{B-\alpha}[a_{i,j}^{(1)}(u_i^\alpha - u_j^\alpha) + (1 - a_{i,j}^{(1)})(u_j^\alpha - u_i^\alpha) + \\ & a_{i,j}^{(2)}(v_i^\alpha - v_j^\alpha) + (1 - a_{i,j}^{(2)})(v_j^\alpha - v_i^\alpha) - a_{i,j}^{(3)}(u_i^\alpha - v_j^\alpha) - \\ & a_{i,j}^{(4)}(u_j^\alpha - v_i^\alpha)] - K a_{i,j}^{(3)} a_{i,j}^{(4)}. \end{aligned} \tag{17}$$

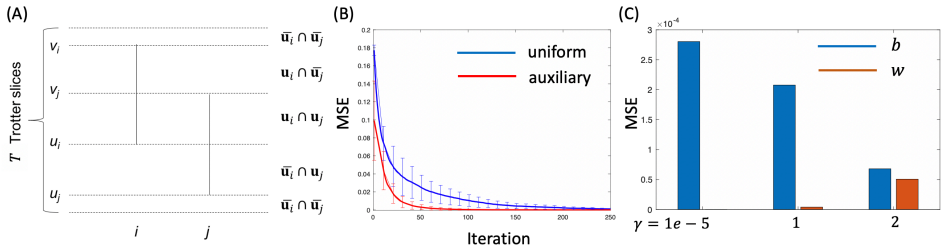

Figure 3: Auxiliary energy for restricted world-lines. (A) provides a schematic of the notation used in App. A.1. (B) compares the efficiency of importance sampling methods for estimating the expected bias statistics in a partially-clamped QBM model using the auxiliary energy introduced in App. A.1 versus uniform sampling as the proposal distribution. (C) compares the accuracy of continuous-time MCMC with population annealing for estimating both bias and pairwise weight statistics in QBMs with different γ terms.

The auxiliary variables here represent the conditions $a^{(1)}_{i,j} = [u_i > u_j]$, $a^{(2)}_{i,j} = [v_i > v_j]$, $a^{(3)}_{i,j} = [u_i > v_j]$, $a^{(4)}_{i,j} = [u_j > v_i]$. When these conditions are met, the potential represents the correct energy, according to Eq. 13. Assuming $(w^{(2)}_{i,j} - w^{(1)}_{i,j}) > 0$, $a^{(1)}_{i,j}$ and $a^{(2)}_{i,j}$, will tend to take the correct values, since the first four terms in the summation over α will minimize the energy for these settings. However, the settings of $a^{(3)}_{i,j}$ and $a^{(4)}_{i,j}$ will tend to violate the conditions above. Conversely, when $(w^{(2)}_{i,j} - w^{(1)}_{i,j}) < 0$, the reverse will be true. We discuss below how to address this issue using importance sampling.

With the binary energy so defined, we have that, for $K = \infty$ and $c = 0$:

$$E_2(\mathbf{u}) = E_3(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a}) + C \tag{18}$$

where $\mathbf{u}_{\text{bin}}$, $\mathbf{v}_{\text{bin}}$ are the binarized representation of $\mathbf{u}$, for all configurations in which the following conditions hold for all i and $(i,j) \in G$:

$$\begin{aligned} v_i &\geq u_i \\ a^{(1)}_{i,j} &= [u_i > u_j] \\ a^{(2)}_{i,j} &= [v_i > v_j] \\ a^{(3)}_{i,j} &= [u_i > v_j] \\ a^{(4)}_{i,j} &= [u_j > v_i] \end{aligned} \tag{19}$$

A possible approach to sampling from $P_2(\mathbf{u})) \propto \exp(-E_2(\mathbf{u}))$ is thus to sample from $E_3(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a})$, and reject all samples violating any of the conditions in Eq. 19. As $K \to \infty$, this approach will approximate drawing samples from the true distribution arbitrarily closely. However, as discussed above, the specific values of the unary and pairwise terms will cause certain constraints to be rarely satisfied, meaning that for a large model, the accepted proportion of samples will be low. To increase the proportion of accepted samples, we fine-tune the offset c discussed above in Eq. 16 to reach a desired acceptance ratio, and set $a^{(3)}_{i,j} = a^{(4)}_{i,j} = 0$ for pairwise potentials where $(w^{(2)}_{i,j} - w^{(1)}_{i,j}) > 0$ in Eq. 17 (dropping constraints 4-5 in Eq. 19), and $a^{(1)}_{i,j} = a^{(2)}_{i,j} = 0$ otherwise (dropping constraints 1-2 in Eq. 19). Since the accepted samples are now from a modified energy function, we use importance sampling to estimate the desired expectations of samples drawn from this distribution (Eq. 6); hence, for a given sample, we calculate $\tilde{r} = \tilde{p}/\tilde{q}$, where $\tilde{p}$ and $\tilde{q}$ are the unnormalized true ($\exp(-E_2(\mathbf{u}))$) and approximating ($\exp(-E_3(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a}))$) distributions respectively, for samples drawn from q. For a given function f and samples $1...S$, and letting $\tilde{r}_{\text{tot}} = \sum_s \tilde{r}_s$, we then have:

$$E_p[f(x)] \approx \frac{1}{S} \sum_s \frac{\tilde{r}_s}{\tilde{r}_{\text{tot}}} f(x). \tag{20}$$

The efficiency with which the required gradient statistics can be calculated will depend on a trade-off between the acceptance probability, and the rate of convergence of the approximation in Eq. 20. In Fig. 3B, we show that the constant c may be tuned to enable efficient estimation of the bias update statistics from Eq. 6, as compared to a baseline importance sampling approach, which uses uniform sampling as the approximate distribution q in place of the auxiliary energy E_3. Here, we calculate updates for a 2-spin QBM, having $\{b_1, b_2, w, \gamma\} = \{1, -0.5, 0.5, 4\}$ and $T = 4$ Trotter slices, and set $c = 1$ (performance is evaluated over 500 trials drawn from each clamped configuration). In Fig. 3C, we show that the discretized auxiliary energy

approach is able to reach a similar level of error in estimating the gradient statistics to a continuous-time MCMC approach based on population annealing [11], where the statistics in Fig. 3C are estimated from a QBM over 5 qubits, where we sample bias terms uniformly between -0.5 and 0.5, generate a fully connected weight matrix W with weights uniformly sampled between 0 and 5, and set γ as shown in Fig. 3C. As noted in the main paper, the space complexity of Eq. 15 is $O(N(\log(T)+D))$, where $|G| \leq ND/2$, assuming each node is connected to at most D others.

Finally, we note that the auxiliary energy construction above may be generalized to allow an arbitrary number of break-points. To do so, we form M groups of T Trotter slices. Each group m has its own break-points $0 \leq u_{i,m} \leq v_{i,m} < T$. However, the break-points in neighboring groups m and $m+1$ may be connected by setting $v_{i,m} = T-1$ and $u_{i,m+1} = 0$. Hence, there will be at most $2M$ break-points per world-line, although some of these may be joined, or not present, and each group of Trotter slices can contain at most 2 break-points. The auxiliary energy may then be written:

$$E_4(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a}) \quad = \quad -\sum_{i,m,\alpha} \psi_i(u^\alpha_{i,m}, v^\alpha_{i,m}, a_{i,m}) - \sum_{(i,j)\in G,\alpha} \psi_{i,j}(u^\alpha_{i,m}, u^\alpha_{j,m}, v^\alpha_{i,m}, v^\alpha_{j,m}, a^{(1\ldots4)}_{i,j,m}) \quad (21)$$

where:

$$\psi_i(u^\alpha_{i,m}, v^\alpha_{i,m}, a_{i,m}) \quad = \quad a^{1a}_{i,m} \cdot 2^{B-\alpha}(b^u_i + c_i)(v^\alpha_{i,m}\alpha - u^\alpha_i) + (1 - a^{1a}_{i,m})(2\gamma'_i - K(\tilde{u}^\alpha_{i,m} + \tilde{v}^\alpha_{i,m})) + [m > 0] \cdot a^{1b}_{i,m}[K(B + u^\alpha_{i,m} - v^\alpha_{i,m}) - 2\gamma'_i] \quad (22)$$

where we let $\tilde{u}^\alpha_{i,m} = (1 - u^\alpha_{i,m})$ for $\alpha = 1$, and $u^\alpha_{i,m}$ otherwise (and similarly for $\tilde{v}^\alpha_{i,m}$). The pairwise terms are are identical to Eq. 17, with subscript m's added to all break-point and auxiliary variables. Similarly to the 2 break-point case, when $K = \infty$ and $c = 0$ we have $E_2(\mathbf{u}) = E_4(\mathbf{u}_{\text{bin}}, \mathbf{v}_{\text{bin}}, \mathbf{a}) + C$ when the conditions of Eq. 19 are satisfied for each Trotter slice group. The space complexity of Eq. 21 is $O(NM(\log(T)+D))$; since the full Trotter expansion requires space $O(NT)$, this representation is therefore only efficient if $M << T$.

A.2 Training using Score-Function Gradients and Likelihood Evaluation

We briefly discuss here how models in our HSN framework may be optimized using score-function gradient, and reparameterization gradient methods. For our experimentation, we use score function gradients to calculate the updates in Eq. 6 [14,22]. The derivatives in Eq. 6 with respect to the model parameters θ are straightforward to estimate by drawing samples from the clamped and free distributions either using the full Trotter expansion for QBMs, or our auxiliary energy formulation (see App. A.1). To update the parameters of the variational distributions, $Q_{\phi_l}(\mathbf{z}_l|\mathbf{x})$ we use the score-function estimator:

$$\nabla_{\phi_l}\mathcal{L}_{\text{ELBO}} \quad = \quad \frac{1}{S}\sum_s (\log P_\theta(\mathbf{x}, \mathbf{z}^{(s)}_l) - \log Q_\phi(\mathbf{z}^{(s)}_l|\mathbf{x})) \times \nabla_{\phi_l} \log Q_\phi(\mathbf{z}^{(s)}_l|\mathbf{x}), \quad (23)$$

where $\mathbf{z}^{(s)}_l$, $s = 1\ldots S$ are samples drawn from $Q_{\phi_l}(\mathbf{z}_l|\mathbf{x})$ (for which we use a product of Bernoulli distributions for QBM layers, whose expectation is determined by a neural network parameterized by ϕ). As noted in [14], the distribution $\log P_\theta$ need only be evaluated in up to a normalizing constant; hence, we use log of the expected value of $\exp(-E_{\text{QBM-sc}}(\mathbf{z}^{(s)}_l))$, over samples drawn from $E_{\text{QBM-sc}}$, which is the partially-clamped energy from Eq. 9.

An alternative possibility for optimization would be to use the reparametrization formulation for BM and QBM hidden variables introduced in [9]. We note that [9] uses the reparameterization scheme to calculate both ELBO and Q-ELBO gradients, where the ELBO gradients are calculated using the full Trotter expansion. Hence, our auxiliary energy may be directly substituted in the ELBO updates, avoiding the need to resort to the Q-ELBO approximation, and providing an efficient alternative for QVAE training which allows the local γ terms to be trained concurrently with the other model parameters.

Finally, we outline our approach to estimating the likelihood of a binary output vector with respect to the HSN model as a whole, using Approximate Bayesian Computation (ABC, see [23]). To do so, we set a deviation threshold, δ, and sample a fixed number S of output vectors $\mathbf{x}_s$ from the HSN (by sampling from all latent variables and finally from the output layer). We then estimate the likelihood of a given output $\mathbf{x}_0$ as the fraction of samples which are within a Hamming distance δ of the observed vector:

$$P(\mathbf{x}_0) \approx \frac{\sum_s [D(\mathbf{x}_0, \mathbf{x}_s) \leq \delta]}{S}. \quad (24)$$

When $\delta = 0$ this reduces to the standard frequentist estimator. Throughout our experimentation, we take $\delta = 2$.

A.3 Synthetic model for γ-term interpretation

Finally, we provide details of a synthetic investigation to support the claim in Sec. 2.3 that the γ-terms in the QBM layers provide a tunable mechanism for introducing higher-order interactions into a BM energy.

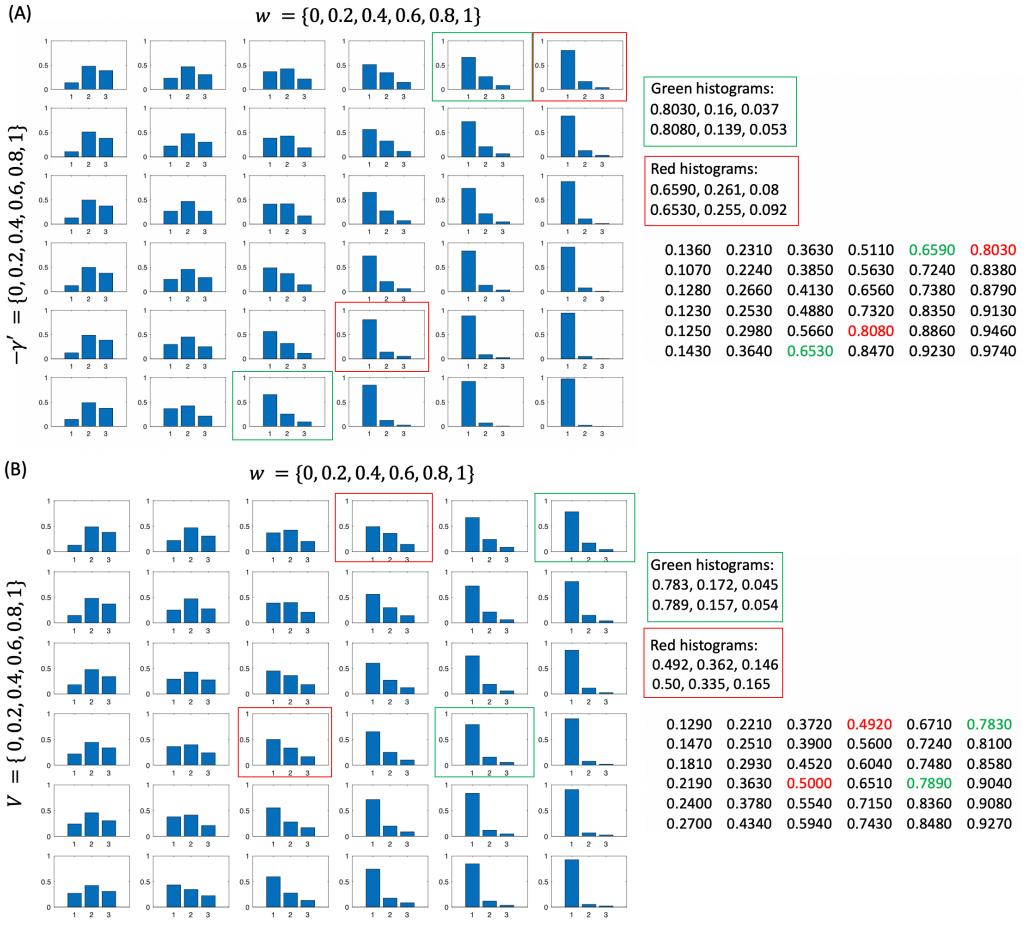

Figure 4: Synthetic model for γ-term interpretation. (A) and (B) show histograms of outputs evaluated using the QBM and higher-order BM energies defined in Eqs. 25 and 26 respectively for varying settings of the model parameters (1000 trials for each configuration). Increasing $-\gamma'$ in (A) is shown to have a similar effect to increasing the weight of the higher-order term V in (B) by comparing models with similar weight on the lowest-energy configuration. See App. A.3 for further details and interpretation.

Particularly, we investigate the following 4-qubit pairwise QBM energy:

$$H = -\sum_i \gamma \sigma_i^{(x)} - \sum_{i \neq j} w \sigma_i^{(z)} \sigma_j^{(z)}, \tag{25}$$

in which we note that the γ parameter is chosen to be the same across all the qubits. We compare this to the following classical BM, with an additional 4-way higher-order term:

$$E(\mathbf{x}) = -\sum_{i \neq j} w[x_i = x_j] + \sum_{i \neq j} w[x_i \neq x_j] - V[x_1 = x_2 = x_3 = x_4]. \tag{26}$$

Due to symmetry, both of these energy functions can be summarized by the probability that a given output is in one of three possible equivalence classes: (1) all outputs identically 0 (resp. 1); (2) three outputs are 0 (resp. 1) and one is 1 (resp. 0); (3) two outputs are 0 and two 1. All outputs in a given equivalence class have the same probability. Further, when $\gamma = 0$ and $V = 0$, the two models agree. In Fig. 4, we plot the histograms for the three equivalence classes above as we vary γ and V for the first (A) and second (B) model respectively. We are interested in comparing models with approximately the same probability for class (1) when $\gamma = 0$ and $V = 0$, versus when either of these terms is non-zero. The numerical grids on the lower-right pick out pairs of histograms for which this is the case, and the exact numerical values for these histograms are given in the colored boxes (the $\gamma = 0/V = 0$ case is given first). As can be seen, increasing $-\gamma'$ or V has the similar effect of decreasing the frequency with which equivalence class (2) is observed, and increasing class (3) for a fixed

probability of class (1). We may interpret this as increasing the 'stability' of the lowest energy configurations (i.e. those in class (1)): by reducing the probability mass in class (2), a system at the lowest energy is more resilient to single bit-flips away from this configuration. In this way, increasing $-\gamma'$ in this toy problem has a similar effect to explicitly including an extra higher-order potential in a classical BM to enforce the lowest energy configuration. We suggest that in larger systems, the local γ_i terms may similarly reinforce the joint configurations of restricted groups of outputs by increasing the coupling magnitudes ($-\gamma_i'$) between the Trotter slices for outputs participating in higher-order interactions. We note, however, that the precise effect will depend on the relative strengths of the γ_i and w_{ij} terms, as well as the differences between the γ_i terms.