
QUANTUM-INSPIRED STRUCTURE-AWARE DIFFUSION FOR EFFICIENT 3D MOLECULAR GENERATION

Anonymous authors

Paper under double-blind review

ABSTRACT

The high computational cost of classical diffusion models can limit their use in large-scale 3D molecular generation for drug and material discovery. We introduce **Structure-aware Quantum Diffusion (SQ-Diff)**, the first full quantum diffusion model for this task, designed to leverage potential quantum advantages in the Noisy Intermediate-Scale Quantum era. Structural priors (e.g., inter-atomic distances) are encoded into the initial quantum state via a novel state preparation procedure that yields a unified normalization scheme dependent only on the number of atoms. The denoising process is driven by a Quantum U-Net, a fully quantum architecture that combines learnable variational quantum circuits with parameter-free operators. Training is guided by these structural priors enforced through a graph-based objective function to maintain structural consistency. Experimentally, SQ-Diff generates valid and diverse 3D molecules and shows improved performance over existing quantum-based methods. While a gap in generation quality compared to leading classical models remains, our model matches the inference speed of the fastest classical approaches with only a few quantum parameters, setting a new benchmark for pure quantum generative models.

1 INTRODUCTION

The automated discovery of novel molecules with desired 3D structures is a key area of modern drug discovery and material science (Sanchez-Lengeling & Aspuru-Guzik, 2018). By generating valid and diverse molecules, computational methods can significantly accelerate the design-test-analyze cycle, reducing costs and exploring vast regions of the chemical space that are otherwise inaccessible. Among various deep generative models, such as Variational Autoencoders (VAEs) that learn a latent representation of the data (Kingma & Welling, 2013) and Generative Adversarial Networks (GANs) that use a discriminator to refine generated samples (Goodfellow et al., 2014), diffusion models (Ho et al., 2020; Song et al., 2021b) have recently emerged as a powerful paradigm for 3D molecular generation. Their performance stems from their formulation as score-based models that learn the gradient of the data log-likelihood, allowing them to capture fine-grained details of the underlying distribution. This has led to state-of-the-art performance in generating high-quality molecular geometries (Hoogetboom et al., 2022; Xu et al., 2023). However, despite their success, classical diffusion models face persistent challenges. The iterative denoising process is computationally intensive, leading to slow sampling speeds that can hinder high-throughput virtual screening (Wang et al., 2025). Furthermore, generative models can struggle to produce novel scaffolds beyond what is represented in their training data, a well-documented challenge in benchmarking de novo molecular design (Brown et al., 2019).

Quantum computing offers a fundamentally different paradigm for generative modeling, holding the potential to overcome some of these classical limitations (Biamonte et al., 2017; Smaldone et al., 2025). The principles of quantum mechanics, such as superposition and entanglement, provide access to an exponentially large Hilbert space. This vast state space is particularly well-suited for representing the intricate conformational and electronic states of molecules, which are themselves quantum systems. This allows quantum models to represent and explore complex, high-dimensional probability distributions with potentially greater efficiency and expressivity than their classical counterparts (Cerezo et al., 2021). Recent work has already demonstrated that hybrid quantum-classical models can generate novel, synthesizable molecules that are validated in the lab, showcasing the potential of this paradigm to impact real-world drug discovery campaigns (Ghazi Vakili et al., 2025).

054 Despite this promise, the application of Quantum Machine Learning (QML) to molecular generation
055 is still nascent (Smaldone et al., 2025). Early efforts, such as quantum GANs with a hybrid gener-
056 ator (QGAN-HG) (Li et al., 2021) and scalable quantum generative autoencoders (SQ-VAE) (Li &
057 Ghosh, 2022), relied on hybrid quantum-classical architectures, focusing on 2D molecular graphs.
058 The more challenging task of direct 3D molecular generation in a pure quantum framework remains
059 largely unexplored. While the recently proposed quantum circuit-based VAE for 3D molecular gener-
060 ation (QVAE-Mole) (Wu et al., 2024) represents a step forward by using a pure quantum VAE
061 for 3D generation, limitations persist. These models often use state preparation procedures that do
062 not fully embed the graphical topology, particularly inter-atomic distances. Failing to encode this
063 geometric information can lead to structurally invalid outputs. Moreover, graph priors such as per-
064 mutation invariance are not explicitly considered (Zhao et al., 2024). VAE-based approaches can
065 also be prone to posterior collapse (Bowman et al., 2016). These limitations highlight the need for a
066 more robust and structurally-aware quantum generative framework.

067 To address these challenges, we introduce, to our knowledge, the first pure quantum diffusion model
068 for de novo 3D molecular generation. Our approach integrates molecular graph structure with a
069 quantum diffusion process. We propose a novel state preparation procedure that encodes atom and
070 bond information into the initial quantum state, creating a latent representation that reflects the
071 molecule’s topology. The core of our generative process is a quantum denoising U-Net (Ronneberger
072 et al., 2015) that learns to reverse the noise process. This architecture is composed of variational
073 quantum circuits (VQCs) for learnable transformations and parameter-free quantum operations for
074 downsampling and upsampling. The training is supervised by structural priors, including node-edge
075 consistency and permutation invariance, to guide the model toward generating valid structures. For
076 efficient inference, we adapt the Denoising Diffusion Implicit Models (DDIMs) (Song et al., 2021a)
077 sampling scheme.

078 **Our main contributions are summarized as follows:**

- 079 1. We propose the first pure quantum diffusion model for 3D de novo molecular generation, featur-
080 ing a Quantum U-Net architecture that uses VQCs for learnable denoising and parameter-free
081 operators for multi-scale feature processing.
- 082 2. We introduce a state preparation method that explicitly encodes atom and bond information from
083 molecular graphs, combined with training supervised by structural priors to guide the generative
084 process.
- 085 3. We empirically demonstrate on the QM9 benchmark that our model achieves inference speeds
086 comparable to the fastest classical methods and shows improved generation quality over promi-
087 nent quantum models, presenting a promising application of QML in scientific discovery.

089 2 RELATED WORK

090
091 Our work is situated at the intersection of classical generative models for molecules and the emerging
092 field of QML.

093 **Classical Generative Models for Molecules.** Deep generative models have significantly advanced
094 molecule discovery. Early approaches were dominated by VAEs (Kingma & Welling, 2013) and
095 GANs (Goodfellow et al., 2014). More recently, diffusion models (Ho et al., 2020) have become the
096 state of the art, particularly for generating 3D molecular structures. Models like Geometric Diffu-
097 sion Model (GeoDiff) (Xu et al., 2022) and Equivariant Graph Diffusion Model (EDM) (Hoogeboom
098 et al., 2022) demonstrated strong performance by leveraging equivariant neural networks to respect
099 the geometric symmetries of molecules. These models typically operate on continuous representa-
100 tions of atom coordinates and types, progressively denoising them from a simple prior distribution.
101 Key application areas for these models include de novo generation (Hoogeboom et al., 2022), con-
102 former generation (Jing et al., 2022), structure-based drug design (Schneuing et al., 2024), and
103 molecular docking (Corso et al., 2023).

104 **Accelerating Classical Diffusion Models.** A significant drawback of traditional diffusion mod-
105 els is their slow sampling speed, which requires hundreds or thousands of sequential denoising
106 steps. To address this, DDIMs (Song et al., 2021a) were introduced, enabling faster generation
107 through a deterministic, non-Markovian reverse process. This is achieved by re-formulating the re-
verse step to depend on the predicted clean data and the current noisy sample, allowing for larger,

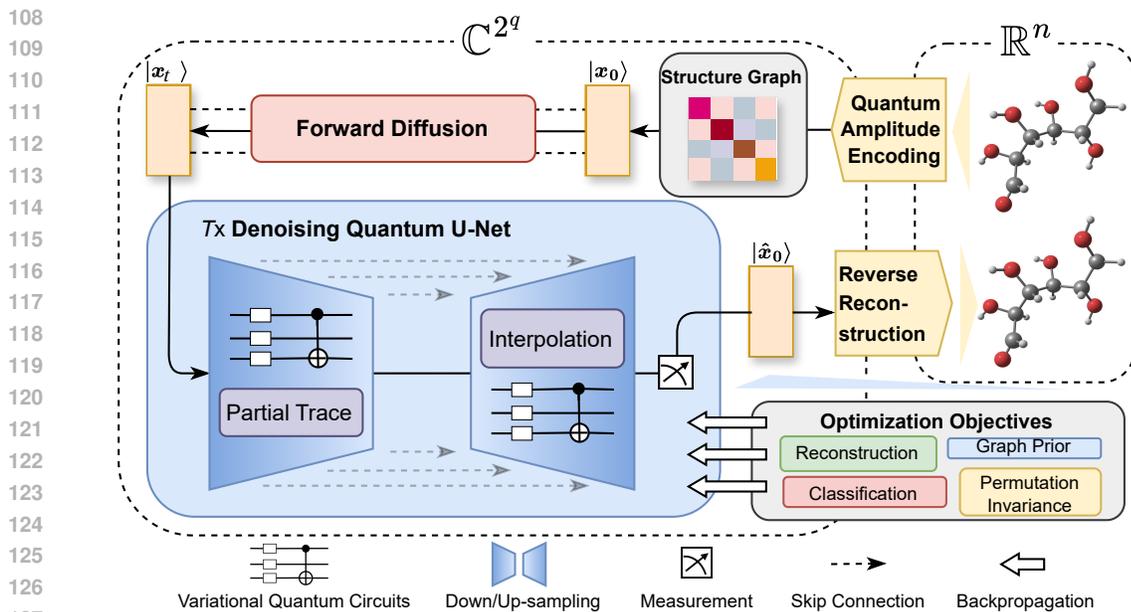


Figure 1: **The workflow of our SQ-Diff model.** A 3D molecular graph is encoded into an initial quantum state $|x_0\rangle$. The forward diffusion process iteratively adds noise, producing a noisy state $|x_T\rangle$. To reverse this, a Quantum U-Net, built from variational quantum circuits and parameter-free down/up-sampling operators, is trained to denoise the state at each timestep. The network is guided by objectives including reconstruction, classification, and graph-based structural priors. During inference, the trained model generates molecular structures from random noise.

more direct jumps along the denoising trajectory. More recently, a new class of models based on flow matching and optimal transport has emerged, offering further acceleration. Geometric Optimal Transport (GOAT) (Hong et al., 2025), for example, formulates molecular graph generation as a time-continuous flow-matching problem, enabling high-quality generation in fewer steps.

Quantum Generative Models. QML offers a promising alternative for generative tasks due to the high expressive power of VQCs (Cerezo et al., 2021). Initial forays into molecular generation, such as QGAN-HG (Li et al., 2021) and SQ-VAE (Li & Ghosh, 2022), were primarily hybrid quantum-classical models focused on generating 2D molecular graphs. Subsequent advancements have expanded this landscape with models like MolGAN with VQC (Kao et al., 2023), which explores quantum GANs (Dallaire-Demers & Killoran, 2018) for generative chemistry, hybrid quantum-classical architectures for drug design (Gircha et al., 2023), and quantum-inspired tensor network approaches for small molecular datasets (Moussa et al., 2023). More recent developments include hybrid quantum cycle GANs for small molecule generation (Anoshin et al., 2024), quantum computing-based deep learning for molecular design optimization (Ajagekar & You, 2023), and quantum-enhanced generative models targeting specific inhibitors like KRAS (Ghazi Vakili et al., 2025). The development of pure quantum models for direct 3D generation is more recent and challenging. QVAE-Mole (Wu et al., 2024) was the first to propose a pure quantum VAE for this task. To date, a pure quantum diffusion model for 3D molecular generation has not been proposed, leaving a significant gap that our work aims to fill.

3 METHODS

In this section, we present the technical details of our proposed Structure-aware Quantum Diffusion (SQ-Diff) model. We first present our novel state preparation method for encoding 3D molecular structures into quantum states. We then describe the pure Quantum U-Net architecture that drives the denoising process. Finally, we introduce the graph-based optimization objectives designed to guide the generation of valid molecules. The workflow is shown in Figure 1. See Appendix A for the quantum basics used in this paper.

3.1 QUANTUM STATE PREPARATION FOR MOLECULE GRAPHS

The foundation of our generative process is a novel state preparation procedure that maps a molecule’s 3D geometry and atomic features into a quantum state, which is fundamentally different from the classical encoding approach as shown in Figure 2. We begin with the molecule’s point cloud data, where each of the N atoms is described by its 3D coordinates $\mathbf{a}_i = (a_{ix}, a_{iy}, a_{iz}) \in \mathbb{R}^3$ and a one-hot vector \mathbf{t}_i for its atom type. As a preprocessing step, we translate by the per-axis minimum and scale to $[0, 1]$: letting $\mathbf{a}_{\min} = (\min_N a_{ix}^{ori}, \min_N a_{iy}^{ori}, \min_N a_{iz}^{ori})$ and $D = \max_{1 \leq i, j \leq N} \|\mathbf{a}_i^{ori} - \mathbf{a}_j^{ori}\|_\infty$, we set $\mathbf{a}_i \leftarrow (\mathbf{a}_i^{ori} - \mathbf{a}_{\min})/D$. This choice matches the derivation of the unified normalization in Appendix C.1.

Feature Encoding. Our method constructs a molecular structure graph M that embeds structural information for the subsequent amplitude encoding. The node-level features, positioned along the diagonal, consist of three components: 1) atom coordinates \mathbf{a}_i and 2) atom type one-hot vectors \mathbf{t}_i , which encode basic node information, and 3) auxiliary values $\sqrt{(2N+1)(3-\|\mathbf{a}_i\|_2^2)}$ inspired by prior work (Wu et al., 2024; Rathi et al., 2023), which facilitate the subsequent normalization. The edge-level features, located in the off-diagonal positions, comprise the Euclidean distance $\|\mathbf{a}_i - \mathbf{a}_j\|_2$ ($i \neq j$), which is inherently E(3)-invariant (Garcia Satorras et al., 2021).

Normalization Scheme. A key innovation of our work is a state preparation procedure that yields a unified normalization scheme dependent only on the number of atoms. We introduce a specific data-dependent correction term v_{corr} and engineer the feature set such that the squared L2-norm of the flattened feature vector \mathbf{m} simplifies to a concise form (see Appendix C.1 for full derivation):

$$\|\mathbf{m}\|_2^2 = 6N^2 + 4N \quad (1)$$

This result provides a unified normalization paradigm: all entries are normalized by dividing by $\sqrt{6N^2 + 4N}$, which allows the model to process molecules of varying sizes on an equal footing (The form of the initial quantum state is given in Appendix C.1).

The reverse process, **reverse reconstruction**, which maps the final denoised quantum state back to a 3D structure, is the exact symmetric inverse of this preparation step. As it is mathematically straightforward and deterministic, we will not elaborate on it further in a separate section.

3.2 QUANTUM FORWARD DIFFUSION PROCESS

The forward diffusion process gradually injects noise into the initial quantum state $|x_0\rangle$ over a sequence of T timesteps. We represent the quantum state $|x_0\rangle$ by its vector of amplitudes, $\mathbf{x}_0 \in \mathbb{C}^{2^q}$. The process is defined as a fixed Markov chain that produces a sequence of increasingly noisy latent amplitude vectors $\mathbf{x}_1, \dots, \mathbf{x}_T$.

Following standard diffusion models (Ho et al., 2020), the transition at each step t is modeled as a Gaussian kernel with a cosine variance schedule $\{\beta_t \in (0, 1)\}_{t=1}^T$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \quad (2)$$

A notable property of this process is that we can sample \mathbf{x}_t at an arbitrary timestep t directly from the initial state \mathbf{x}_0 in a closed form. Using the notation $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$, the

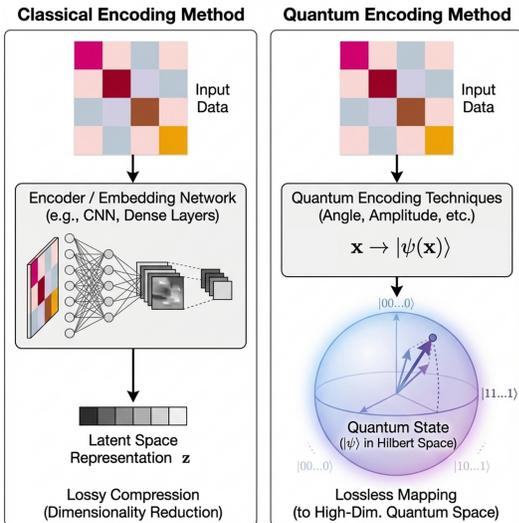


Figure 2: **Comparison of different decoding methods.** The left panel shows the classical decoding method. The right panel shows the quantum decoding method.

Algorithm 1 SQ-Diff Training Procedure

1: **Initialize:** Quantum U-Net parameters θ , optimizer.
2: **repeat**
3: Sample a batch of molecules $\{M_i\}$ from the training dataset \mathcal{D} .
4: **for** each molecule M in the batch **do**
5: $\mathbf{x}_0 \leftarrow \text{StatePreparation}(M)$ ▷ Encode molecule into an amplitude vector
6: $t \sim \text{Uniform}(\{1, \dots, T\})$ ▷ Sample a random timestep
7: $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ▷ Sample Gaussian noise
8: $\mathbf{x}_t \leftarrow \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ ▷ Apply forward diffusion process
9: $\mathbf{x}_t \leftarrow \mathbf{x}_t / \|\mathbf{x}_t\|_2$ ▷ Normalize to get a valid quantum state vector
10: $|x_t\rangle \leftarrow \text{PrepareStateFromVector}(\mathbf{x}_t, t)$ ▷ Embed time and prepare input state
11: $|\hat{x}_0\rangle \leftarrow \text{QuantumUNet}(|x_t\rangle, t; \theta)$ ▷ Predict denoised state with Q-U-Net
12: $\hat{\mathbf{x}}_0 \leftarrow \text{GetAmplitudes}(|\hat{x}_0\rangle)$
13: Calculate the total loss $\mathcal{L} \leftarrow \mathcal{L}_{\text{recon}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{prior}}\mathcal{L}_{\text{prior}} + \lambda_{\text{perm}}\mathcal{L}_{\text{perm}}$
14: **end for**
15: Update parameters: $\theta \leftarrow \text{OptimizerStep}(\theta, \nabla_{\theta}\mathcal{L})$
16: **until** convergence criteria are met

distribution of \mathbf{x}_t conditioned on \mathbf{x}_0 is given by:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1 - \bar{\alpha}_t)\mathbf{I}) \quad (3)$$

This allows for efficient sampling of a noisy vector at any step t via the reparameterization trick:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, \quad \text{where } \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (4)$$

To inform the denoising network of the current noise level, we embed the timestep t using the standard sinusoidal positional encoding (Vaswani et al., 2017), creating a time embedding vector \mathbf{e}_t . This embedding is added to the noisy state vector \mathbf{x}_t after a simple fixed projection to match dimensions. This vector serves as the input amplitude vector for the quantum state $|x_t\rangle$ that is fed into our quantum denoising U-Net.

3.3 QUANTUM DENOISING PROCESS

Consistent with the paradigm of classical diffusion models, the reverse process requires a learnable neural network trained to denoise the corrupted state at each timestep. To effectively learn the complex interactions between features at different resolutions, we employ a U-Net-style architecture (Ronneberger et al., 2015), a design empirically validated for its multi-scale analysis capabilities. Our architecture is constructed from a series of VQCs. To ensure the model remains fully within the quantum domain, we introduce parameter-free downsampling and upsampling operators based on partial trace (Romero et al., 2017) and interpolation, respectively, which avoids introducing any classical learnable parameters. During the inference phase, the trained Quantum Denoising U-Net utilizes a DDIM sampling strategy for efficient de novo generation of 3D molecules. Algorithm 1 outlines the end-to-end training process. Algorithm details for generation are provided in Appendix B.

3.3.1 ARCHITECTURE OF THE QUANTUM U-NET

We propose a pure quantum-parameterized denoising U-Net as shown in Figure 3. Analogous to its classical counterpart, our network comprises a downsampling path, an upsampling path, a bottleneck layer, and skip connections that bridge corresponding levels of the two paths. The downsampling path progressively reduces the dimensionality of the quantum state through a series of n blocks. Each block consists of a quantum ansatz $U_{d_i}(\theta_{d_i})$ followed by a downsampling operator D_i . Given an initial input state $|x_t\rangle$, the state at the end of the downsampling path, $|z_t\rangle$, is produced as follows:

$$|z_t\rangle = \left(\prod_{i=1}^n U_{d_i}(\theta_{d_i}) D_i \right) |x_t\rangle \quad (5)$$

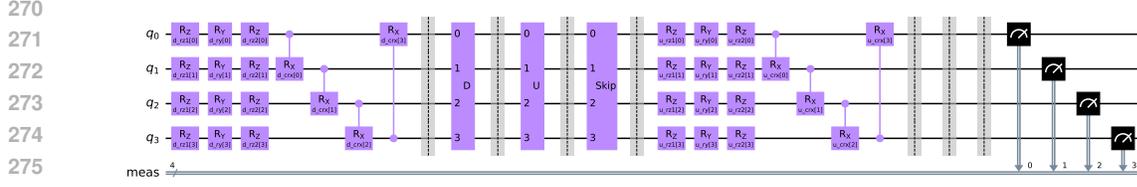


Figure 3: **The structure of one operational block within our Quantum U-Net**, illustrating both the downsampling and upsampling paths. The downsampling path consists of a VQC layer followed by a parameter-free downsampling operator (D). The upsampling path is symmetric, comprising a VQC and an upsampling operator (U), and incorporates information from the corresponding downsampling block via a skip connection.

At the bottleneck, a final quantum ansatz $U_b(\theta_b)$ is applied to this compressed representation to yield the state $|\hat{z}_t\rangle$:

$$|\hat{z}_t\rangle = U_b(\theta_b) |z_t\rangle \quad (6)$$

The upsampling path is structurally symmetric, iteratively expanding the state representation. A key feature is the incorporation of skip connections, which combine the feature map from the previous upsampling layer with the one from the corresponding downsampling layer. This ensures that high-resolution feature information from early stages is available during the reconstruction process. The final predicted clean state, $|\hat{x}_0\rangle$, is generated by a similar product of upsampling operators U_i and ansatzes $U_{u_i}(\theta_{u_i})$:

$$|\hat{x}_0\rangle = \left(\prod_{i=1}^n U_{u_i}(\theta_{u_i}) U_i \right) |\hat{z}_t\rangle \quad (7)$$

Due to the skip connections, the input to the j -th upsampling ansatz is a normalized superposition of the output from the previous upsampling block and the state from the corresponding downsampling block. Denoting the output of the i -th downsampling block as $|x_t^i\rangle$ and the j -th upsampling block as $|\hat{z}_t^j\rangle$, the input is formed as:

$$|\hat{z}_t^j\rangle = \text{Normalize} \left(U_{u_{j-1}}(\theta_{u_{j-1}}) |\hat{z}_t^{j-1}\rangle + |x_t^{n-j+1}\rangle \right) \quad (8)$$

where $|x_t^0\rangle = |x_t\rangle$ and $|\hat{z}_t^n\rangle = |\hat{x}_0\rangle$. Here, $\text{Normalize}(\cdot)$ denotes L2 renormalization to unit norm. The designs of the quantum ansatz and the training-free down/up-sampling operators are detailed in Appendix C, and the training procedure for the Quantum U-Net is provided in Algorithm 1.

3.3.2 DDIM FOR EFFICIENT SAMPLING

Standard Denoising Diffusion Probabilistic Model inference requires thousands of sequential denoising steps, a process that is computationally prohibitive for our quantum framework. We therefore adopt the DDIM formalism (Song et al., 2021a) to accelerate sampling, enabling generation in a fraction of the steps with comparable quality. We set $\bar{\alpha}_0 = 1$ by convention.

The DDIM update to obtain $\mathbf{x}_{t_{\text{prev}}}$ from \mathbf{x}_t can be written as:

$$\mathbf{x}_{t_{\text{prev}}} = \sqrt{\bar{\alpha}_{t_{\text{prev}}}} \hat{\mathbf{x}}_0 + \sqrt{1 - \bar{\alpha}_{t_{\text{prev}}} - \sigma_t^2} \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}} + \sigma_t \boldsymbol{\epsilon}, \quad (9)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}} = (\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \hat{\mathbf{x}}_0) / \sqrt{1 - \bar{\alpha}_t}$, and $\sigma_t = \eta \sqrt{\frac{1 - \bar{\alpha}_{t_{\text{prev}}}}{1 - \bar{\alpha}_t}} \sqrt{1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t_{\text{prev}}}}}$. The parameter η controls sampling stochasticity. See Algorithm 2 for random generation details.

For conditional generation based on a specific property y , we adopt a classifier-guidance strategy (Dhariwal & Nichol, 2021). Given a pre-trained classifier r_ϕ that predicts y from a clean-state estimate, we adjust the random prediction by taking a gradient step on a property loss:

$$\hat{\mathbf{x}}_0(\mathbf{x}_t, t) = \hat{\mathbf{x}}_{0, \text{uncond}}(\mathbf{x}_t, t) - \lambda \nabla_{\hat{\mathbf{x}}_0} \mathcal{L}_{\text{prop}}(r_\phi(\hat{\mathbf{x}}_{0, \text{uncond}}(\mathbf{x}_t, t)), y). \quad (10)$$

The guided $\hat{\mathbf{x}}_0$ is then used to compute the corresponding noise for the DDIM update,

$$\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}} = \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \hat{\mathbf{x}}_0}{\sqrt{1 - \bar{\alpha}_t}}, \quad (11)$$

which is substituted into the update rule. This framework allows us to generate high-quality molecules in just 10–50 sampling steps. See Algorithm 3 for conditional generation details.

3.3.3 STRUCTURE-AWARE OPTIMIZATION OBJECTIVES

Our model is trained by minimizing a composite loss function \mathcal{L} designed to ensure the generation of chemically valid and structurally consistent molecules. The overall objective is a weighted sum of four components:

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{prior}}\mathcal{L}_{\text{prior}} + \lambda_{\text{perm}}\mathcal{L}_{\text{perm}} \quad (12)$$

The primary component is the reconstruction loss $\mathcal{L}_{\text{recon}}$, which corresponds to predicting the clean state \mathbf{x}_0 :

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{\mathbf{x}_0, \epsilon, t} \left[\left\| \mathbf{x}_0 - \hat{\mathbf{x}}_0(\sqrt{\alpha_t} \mathbf{x}_0 + \sqrt{1 - \alpha_t} \epsilon, t) \right\|_2^2 \right] \quad (13)$$

To enhance the accuracy of atom type prediction, we incorporate a classification loss \mathcal{L}_{cls} based on the Focal Loss (Lin et al., 2017), which focuses training on hard-to-classify examples. To enforce structural integrity, we introduce a graph prior loss $\mathcal{L}_{\text{prior}}$, which penalizes inconsistencies between reconstructed node features (e.g., coordinates) and edge features (e.g., inter-atomic distances), and ensures that predicted atom-type vectors are valid probability distributions. Finally, because molecular graphs are invariant to node permutations, we include a permutation invariance loss $\mathcal{L}_{\text{perm}}$. This term encourages the model to produce equivalently permuted outputs when the input atoms are re-ordered, thereby embedding this fundamental chemical symmetry into the learning process (Zhou et al., 2020). A detailed formulation of each loss component is provided in Appendix D.

4 EXPERIMENTS

In line with Bausch (2020); Ye et al. (2023); Yan et al. (2023); Wu et al. (2024), we conduct our experiments on classical hardware using simulators from the TorchQuantum library (Wang et al., 2022). Our framework is, however, designed to be executable on NISQ computers, leveraging the library’s compatibility with interfaces for real quantum devices. Further details on the simulator are provided in Appendix G.

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate our model on QM9 (Ramakrishnan et al., 2014), a standard benchmark for molecular generation containing about 130K small molecules with up to nine heavy atoms. The dataset provides 3D atomic coordinates and chemical properties. The objective is to train the model for random and conditional generation of molecular structures, including coordinates and atom types (C, N, O, F). We adopt the data partition from Anderson et al. (2019), which designates 100K, 18K, and 10K for training, validation, and testing respectively. Hardware and hyperparameter settings are summarized in Appendix G.

Baselines. Our comparative analysis includes two main categories of models. The first consists of **classical methods** with no quantum components, including the continuous normalizing flow model E(n) Equivariant Normalizing Flows (E-NFs) (Garcia Satorras et al., 2021), autoregressive models Generative SchNet (G-SchNet) (Gebauer et al., 2019) and Generative SphereNet (G-SphereNet) (Luo & Ji, 2022), diffusion-based models EDM (Hooeboom et al., 2022) and Geometric Latent Diffusion Model (GeoLDM) (Xu et al., 2023), and the flow-matching model GOAT (Hong et al., 2025). The second category consists of **hybrid/quantum methods** that utilize trainable quantum parameters. This group includes hybrid models for 2D molecular graph generation, such as SQ-VAE (Li & Ghosh, 2022) and QGAN-HG (Li et al., 2021) (with its variant P2-QGAN-HG), as well as QVAE-Mole (Wu et al., 2024) (with its variant QVAE-Mole (fid.)), a pure quantum VAE for 3D molecular generation.

Metrics. We assess model performance based on generation quality and efficiency. To evaluate quality, following prior work (Hooeboom et al., 2022), we generate 10K molecules and measure their **Validity (V)**, **Uniqueness (U)**, and **Novelty (N)**. Validity is the percentage of molecules adhering to chemical valency rules. Uniqueness is the ratio of distinct molecules among valid samples. Novelty is the proportion of valid molecules not present in the training set. To penalize models that achieve high uniqueness or novelty by generating invalid structures, we report the composite metrics $\mathbf{V} \times \mathbf{U}$,

$V \times N$, and $V \times U \times N$. For efficiency, we report the number of trainable # C/Q (classical/quantum) parameters and the effective time T_{eff} , defined as the time to generate one valid, unique, and novel molecule.

Table 1: **Performance comparison for random 3D molecule generation on the QM9 dataset.** All models generated 10k molecules. The table is divided into classical and quantum-based methods. Quality metrics (V, $V \times U$, $V \times N$, $V \times U \times N$): higher is better. Efficiency (T_{eff}): lower is better. **Best classical score is in red, and best quantum/hybrid score is in blue.**

Methods	Class	V (%)	$V \times U$ (%)	$V \times N$ (%)	$V \times U \times N$ (%)	# C/Q	T_{eff} (s)
SQ-Diff (Classic) (Subsec. 4.4)	Classic	70.7	3.0	70.7	3.0	0.7M / 0	2.69
E-NFs (Garcia Satorras et al., 2021)	Classic	41.1	40.8	34.9	34.7	0.6M / 0	0.72
G-SchNet (Gebauer et al., 2019)	Classic	82.4	73.3	67.1	59.7	0.9M / 0	0.69
G-SphereNet (Luo & Ji, 2022)	Classic	82.6	29.8	37.8	13.6	3.1M / 0	4.04
EDM (Hooeboom et al., 2022)	Classic	91.9	90.7	75.3	74.3	5.3M / 0	1.68
GeoLDM (Xu et al., 2023)	Classic	93.8	92.7	54.5	53.9	12.5M / 0	1.86
GOAT (Hong et al., 2025)	Classic	92.9	92.0	73.0	72.3	8.6M / 0	0.12
SQ-VAE (Li & Ghosh, 2022)	Hybrid	44.2	7.2	16.3	2.7	128 / 224	5.56
QGAN-HG (Li et al., 2021)	Hybrid	66.6	8.1	18.5	2.2	0.5M / 38	1.82
P2-QGAN-HG (Li et al., 2021)	Hybrid	17.6	12.4	9.5	6.7	0.1M / 14	0.30
QVAE-Mole (Wu et al., 2024)	Quantum	78.1	27.4	57.4	20.1	0 / 602	0.40
QVAE-Mole (fid.) (Wu et al., 2024)	Quantum	74.4	26.9	31.5	11.4	0 / 602	0.70
SQ-Diff (Ours)	Quantum	80.1	48.3	79.9	48.1	0 / 600	0.12

4.2 RANDOM GENERATION

We first evaluate random 3D molecular generation. The results are summarized in Table 1. Our model, SQ-Diff, shows a balance of generation quality and efficiency. Generated examples are visualized in Appendix H.

Performance Comparison with Classical Methods. SQ-Diff sets a new standard for pure quantum generative models and demonstrates a promising direction for quantum-inspired discovery. When compared with classical methods, a gap in sample quality remains; for instance, models like GOAT and GeoLDM achieve higher validity. This is an expected outcome, as Quantum Machine Learning (QML) is still an emerging field. The development of quantum models is constrained by the limitations of Noisy Intermediate-Scale Quantum (NISQ) hardware, which restricts the number of qubits, circuit depth, and connectivity, thereby limiting model size and complexity (Bharti et al., 2022). Consequently, it is a known challenge for current QML models, which are often exploratory proofs of concept, to match the performance of highly optimized classical architectures that leverage millions of parameters (Cerezo et al., 2021). As quantum hardware matures to allow for large-scale, fault-tolerant quantum computers, QML models could see significant performance improvements with scale, similar to the scaling law phenomenon in large language models, potentially enabling capabilities not present in smaller-scale models (Kaplan et al., 2020; Wei et al., 2022).

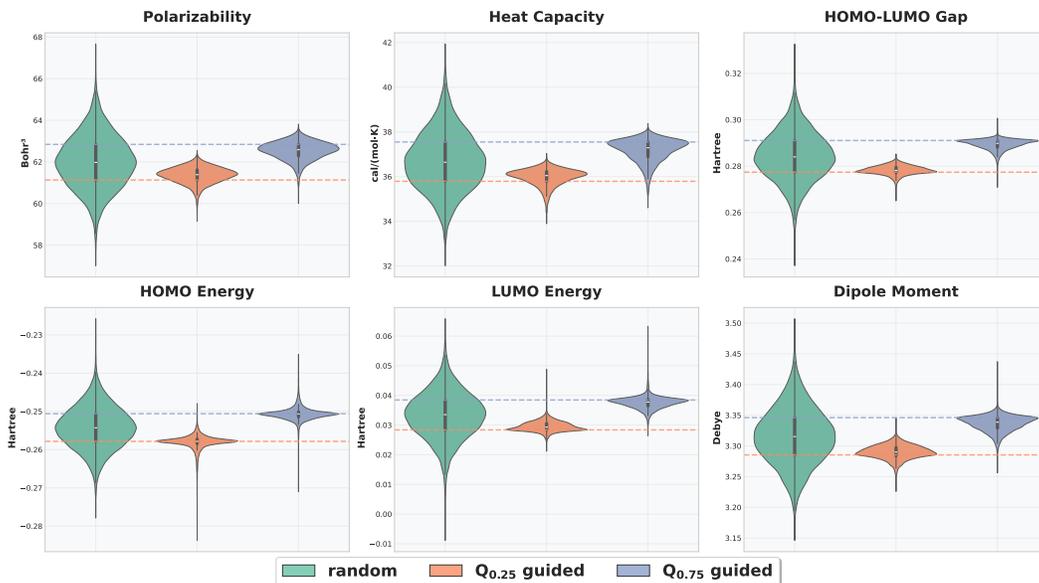
Within this context, the strengths of SQ-Diff are still apparent. A key advantage is its strong capacity for generating novel structures. With a $V \times N$ score of 79.9%, our model surpasses all classical baselines in this metric. This may suggest that the quantum model is less prone to posterior collapse and more effectively explores untapped regions of the chemical space. Most importantly from an efficiency standpoint, SQ-Diff matches the inference time (T_{eff}) of the state-of-the-art flow-matching method GOAT at 0.12s, while being over 10 times faster than established diffusion models like EDM and GeoLDM. This highlights the potential of quantum-inspired diffusion for high-throughput applications.

Performance Comparison with Hybrid/Quantum Methods. Compared to other quantum and hybrid models, SQ-Diff shows a significant improvement in performance. It outperforms existing methods in the composite $V \times U \times N$ metric with a score of 48.1%, more than double that of the next best pure quantum model, QVAE-Mole (20.1%). This improvement is supported by strong performance across all individual metrics. Specifically, SQ-Diff achieves the highest validity (80.1%) and novelty (79.9%) among all quantum-powered models. Its $V \times U$ score of 48.3% indicates greater diversity in the generated samples compared to previous quantum generative models. In terms of

432 efficiency, SQ-Diff’s T_{eff} of 0.12s is on par with the fastest classical methods. It achieves this per-
433 formance using a comparable number of quantum parameters to QVAE-Mole, making it an effective
434 and efficient pure quantum model for 3D molecular generation.
435

436 4.3 CONDITIONAL GENERATION 437

438 To evaluate the controllability of our model, we performed conditional generation experiments tar-
439 geting specific chemical properties. Leveraging the classifier-guidance strategy from Section 3.3.2,
440 we aimed to generate molecules with property values around the lower ($Q_{0.25}$) and upper ($Q_{0.75}$)
441 quartiles of the QM9 dataset’s distribution for six properties (see Appendix E). This approach serves
442 as a clear test of whether the generation process can be steered toward distinct, pre-defined targets,
443 thereby demonstrating fine-grained control. Figure 4 visualizes the results, displaying the prop-
444 erty distributions of 10K molecules generated with guidance toward $Q_{0.25}$ and $Q_{0.75}$, respectively,
445 alongside the distribution from random generation. For all six properties, the guided distributions
446 are clearly shifted and centered around their target quartiles. This demonstrates that our framework
447 provides effective control over the chemical properties of the generated molecules.



467
468 **Figure 4: Distributions of six chemical properties for 10K conditional generated molecules.** For
469 each property, we compare the distribution from random generation (left) with those from condi-
470 tional generation targeting the lower quartile ($Q_{0.25}$, middle) and upper quartile ($Q_{0.75}$, right).
471

472 4.4 ABLATION AND SENSITIVITY ANALYSIS 473

474 **Impact of Quantum Components.** To verify the contribution of the VQCs, we constructed a clas-
475 sical analogue of SQ-Diff by replacing all learnable VQCs with Multi-Layer Perceptrons (MLPs)
476 of a similar design. As shown in Table 1 (SQ-Diff (Classic)), the results reveal a substantial per-
477 formance gap. The classical variant shows a drop in generation quality, with the $V \times U \times N$ metric
478 decreasing from 48.1% to 3.0%. The low uniqueness metric ($V \times U$ of 3.0%) suggests that the model
479 experienced mode collapse. Despite hyperparameter tuning, this issue persisted, suggesting that an
480 architecture designed for VQCs does not translate effectively to a classical setting and may be more
481 prone to local minima. This result indicates that the VQCs are more expressive for this task. Fur-
482 thermore, the classical version is over 20 times slower, with a T_{eff} of 2.69s compared to the quantum
483 model’s 0.12s, highlighting the computational advantages of our quantum design.

484 **Effect of Loss Components.** We performed an ablation study on the loss function components,
485 with results in Table 2. Removing any auxiliary loss term degrades performance. The permutation
invariance term (‘w/o perm’) is the most critical; its removal causes the $V \times U \times N$ score to drop

486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539

Table 2: Ablation study of the loss function components.

Variant	V	V×U	V×N	V×U×N	T _{eff}
Full model	80.1	48.3	79.9	48.1	0.12
w/o cls	63.0	12.3	63.0	12.3	0.49
w/o prior	40.4	14.1	40.4	14.1	0.43
w/o perm	21.3	8.1	21.3	8.1	0.74

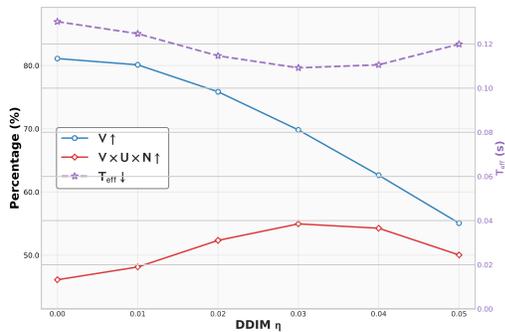


Figure 5: Sensitivity analysis of the DDIM noise parameter η .

to 8.1%, highlighting the importance of encoding chemical symmetry. Removing the graph prior (‘w/o prior’) or classification (‘w/o cls’) losses reduces the metric to 14.1% and 12.3%, respectively, confirming their importance for generating valid and diverse molecules.

Sensitivity to DDIM Noise. We analyzed the model’s sensitivity to the noise parameter (η) in the DDIM sampler, which controls stochasticity. Figure 5 plots the trade-off between generation quality (V×U×N) and sampling efficiency (T_{eff}). The values are provided in Appendix F. Introducing a small amount of noise ($\eta > 0$) improves the V×U×N metric, which peaks around $\eta = 0.03$. This suggests that minor stochasticity helps the model explore the data manifold, enhancing diversity. However, as η increases further, performance declines as excess noise degrades chemical validity.

5 CONCLUSION

In this work, we introduced SQ-Diff, the first pure quantum diffusion model for de novo 3D molecular generation. Our framework uses a novel state preparation procedure and a pure Quantum U-Net guided by a structure-aware loss function. Experiments on the QM9 dataset show that SQ-Diff generates high-quality molecules, outperforming existing quantum and hybrid models. While a quality gap to classical SOTA remains, its inference efficiency is comparable to the fastest classical methods. SQ-Diff operates with a distinct quantum parameterization, establishing a promising direction for quantum-inspired molecular discovery. For the Limitations and Future work, please see Appendix I.

ETHICS STATEMENT

The research presented in this paper is foundational and focuses on the development of a novel quantum algorithm, SQ-Diff, for 3D molecule generation. The potential applications of this work lie in accelerating scientific discovery in fields such as drug design and materials science, which are broadly beneficial to society. We have considered the potential societal impacts and do not foresee any direct or immediate negative consequences or ethical risks associated with this work. We are committed to the responsible dissemination of our research and advocate for the ethical use of quantum computing and artificial intelligence technologies.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our results, all necessary components will be made available. The source code for the SQ-Diff model, environment configurations, and scripts to replicate the experiments reported in this paper are released in an anonymous repository at <https://anonymous.4open.science/r/SQ-Diff-0703>, which will be made public upon publication. We have detailed the primary experimental setup in Section 4. Further comprehensive details regarding the model architecture, training hyperparameters, and dataset processing are provided in Appendix B and Appendix G.

-
- 540 REFERENCES
541
542 Akshay Ajagekar and Fengqi You. Molecular design with automated quantum computing-based
543 deep learning and optimization. *Npj Computational Materials*, 9(1):143, 2023.
- 544 Brandon Anderson, Truong Son Hy, and Risi Kondor. Cormorant: Covariant molecular neural
545 networks. *Advances in neural information processing systems*, 32, 2019.
- 546
547 Matvei Anoshin, Asel Saginalieva, Christopher Mansell, Dmitry Zhiganov, Vishal Shete, Markus
548 Pflitsch, and Alexey Melnikov. Hybrid quantum cycle generative adversarial network for small
549 molecule generation. *IEEE Transactions on Quantum Engineering*, 5:1–14, 2024.
- 550 Adriano Barenco, Charles H Bennett, Richard Cleve, David P DiVincenzo, Norman Margolus, Pe-
551 ter Shor, Tycho Sleator, John A Smolin, and Harald Weinfurter. Elementary gates for quantum
552 computation. *Physical review A*, 52(5):3457, 1995.
- 553 Johannes Bausch. Recurrent quantum neural networks. *Advances in neural information processing*
554 *systems*, 33:1368–1379, 2020.
- 555
556 Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav
557 Anand, Matthias Degroote, Hermanni Heimonen, Jakob S. Kottmann, Tim Menke, Wai-Keong
558 Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. Noisy intermediate-scale quan-
559 tum algorithms. *Reviews of Modern Physics*, 94(1):015004, 2022.
- 560 Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd.
561 Quantum machine learning. *Nature*, 549(7671):195–202, 2017.
- 562
563 Samuel Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio.
564 Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL conference*
565 *on computational natural language learning*, pp. 10–21, 2016.
- 566
567 Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking
568 models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):
569 1096–1108, 2019.
- 570 Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii,
571 Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J Coles. Variational
572 quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.
- 573 Gabriele Corso, Hannes StÅ, Bowen Jing, Regina Barzilay, Tommi Jaakkola, et al. Diffdock: Dif-
574 fusion steps, twists, and turns for molecular docking. In *International Conference on Learning*
575 *Representations*, 2023.
- 576
577 Pierre-Luc Dallaire-Demers and Nathan Killoran. Quantum generative adversarial networks. *Phys-*
578 *ical Review A*, 98(1):012324, 2018.
- 579 Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances*
580 *in neural information processing systems*, 34:8780–8794, 2021.
- 581
582 Victor Garcia Satorras, Emiel Hooeboom, Fabian Fuchs, Ingmar Posner, and Max Welling. E (n)
583 equivariant normalizing flows. *Advances in Neural Information Processing Systems*, 34:4181–
584 4192, 2021.
- 585 Niklas Gebauer, Michael Gastegger, and Kristof Schütt. Symmetry-adapted generation of 3d point
586 sets for the targeted discovery of molecules. *Advances in neural information processing systems*,
587 32, 2019.
- 588
589 Mohammad Ghazi Vakili, Christoph Gorgulla, Jamie Snider, AkshatKumar Nigam, Dmitry
590 Bezrukov, Daniel Varoli, Alex Aliper, Daniil Polykovsky, Krishna M Padmanabha Das, Huel
591 Cox Iii, et al. Quantum-computing-enhanced algorithm unveils potential kras inhibitors. *Nature*
592 *Biotechnology*, pp. 1–6, 2025.
- 593 AI Gircha, AS Boev, K Avchaciov, PO Fedichev, and AK Fedorov. Hybrid quantum-classical ma-
chine learning for generative chemistry and drug design. *Scientific Reports*, 13(1):8250, 2023.

594 Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair,
595 Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information*
596 *processing systems*, 27, 2014.

597 Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in*
598 *neural information processing systems*, 33:6840–6851, 2020.

600 Haokai Hong, Wanyu Lin, and KC Tan. Accelerating 3d molecule generation via jointly geometric
601 optimal transport. In *International Conference on Learning Representations*, 2025.

602 Emiel Hoogeboom, Victor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion
603 for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–
604 8887, 2022.

606 Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional dif-
607 fusion for molecular conformer generation. *Advances in neural information processing systems*,
608 35:24240–24253, 2022.

609 Po-Yu Kao, Ya-Chu Yang, Wei-Yin Chiang, Jen-Yueh Hsiao, Yudong Cao, Alex Aliper, Feng Ren,
610 Alán Aspuru-Guzik, Alex Zhavoronkov, Min-Hsiu Hsieh, et al. Exploring the advantages of
611 quantum generative adversarial networks in generative chemistry. *Journal of Chemical Informa-*
612 *tion and Modeling*, 63(11):3307–3318, 2023.

614 Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child,
615 Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language
616 models. *arXiv preprint arXiv:2001.08361*, 2020.

617 Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint*
618 *arXiv:1312.6114*, 2013.

620 Greg Landrum et al. Rdkit: A software suite for cheminformatics, computational chemistry, and
621 predictive modeling. *Greg Landrum*, 8(31.10):5281, 2013.

622 Junde Li and Swaroop Ghosh. Scalable variational quantum circuits for autoencoder-based drug
623 discovery. In *2022 design, automation & test in europe conference & exhibition (DATE)*, pp.
624 340–345. IEEE, 2022.

626 Junde Li, Rasit O Topaloglu, and Swaroop Ghosh. Quantum generative models for small molecule
627 drug discovery. *IEEE transactions on quantum engineering*, 2:1–8, 2021.

628 Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense
629 object detection. In *Proceedings of the IEEE international conference on computer vision*, pp.
630 2980–2988, 2017.

632 Youzhi Luo and Shuiwang Ji. An autoregressive flow model for 3d molecular geometry generation
633 from scratch. In *International conference on learning representations*, 2022.

634 Charles Moussa, Hao Wang, Mauricio Araya-Polo, Thomas Bäck, and Vedran Dunjko. Applica-
635 tion of quantum-inspired generative models to small molecular datasets. In *2023 IEEE Inter-*
636 *national Conference on Quantum Computing and Engineering (QCE)*, volume 1, pp. 342–348.
637 IEEE, 2023.

638 Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cam-
639 bridge university press, 2010.

640 Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum
641 chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.

642 Lakshika Rathi, Edith Tretschk, Christian Theobalt, Rishabh Dabral, and Vladislav Golyanik. 3d-
643 qae: Fully quantum auto-encoding of 3d point clouds. *arXiv preprint arXiv:2311.05604*, 2023.

644 Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. Quantum autoencoders for efficient
645 compression of quantum data. *Quantum Science and Technology*, 2(4):045001, 2017.

647

648 Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical
649 image segmentation. In *International Conference on Medical image computing and computer-*
650 *assisted intervention*, pp. 234–241. Springer, 2015.

651 Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine
652 learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018.

653 Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du,
654 Carla Gomes, Tom L Blundell, Pietro Lio, et al. Structure-based drug design with equivariant
655 diffusion models. *Nature Computational Science*, 4(12):899–909, 2024.

656 Anthony M Smaldone, Yu Shee, Gregory W Kyro, Chuzhi Xu, Nam P Vu, Rishab Dutta, Marwa H
657 Farag, Alexey Galda, Sandeep Kumar, Elica Kyoseva, and Victor S Batista. Quantum machine
658 learning in drug discovery: Applications in academia and pharmaceutical industries. *Chemical*
659 *Reviews*, 125(12):5436–5460, 2025.

660 Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Interna-*
661 *tional Conference on Learning Representations*, 2021a.

662 Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben
663 Poole. Score-based generative modeling through stochastic differential equations. In *Interna-*
664 *tional Conference on Learning Representations*, 2021b.

665 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,
666 Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information*
667 *processing systems*, volume 30, 2017.

668 Hanrui Wang, Yongshan Ding, Jiaqi Gu, Yujun Lin, David Z Pan, Frederic T Chong, and Song Han.
669 Quantumnas: Noise-adaptive search for robust quantum circuits. In *2022 IEEE International*
670 *Symposium on High-Performance Computer Architecture (HPCA)*, pp. 692–708. IEEE, 2022.

671 Liang Wang, Chao Song, Zhiyuan Liu, Yu Rong, Qiang Liu, and Shu Wu. Diffusion models for
672 molecules: A survey of methods and tasks. *arXiv preprint arXiv:2502.09511*, 2025.

673 Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yo-
674 gatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language
675 models. *arXiv preprint arXiv:2206.07682*, 2022.

676 Huaijin Wu, Xinyu Ye, and Junchi Yan. QVAE-Mole: The quantum VAE with spherical latent
677 variable learning for 3-d molecule generation. In *Advances in Neural Information Processing*
678 *Systems*, volume 37, 2024.

679 Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: a geometric
680 diffusion model for molecular conformation generation. In *International Conference on Learning*
681 *Representations*, 2022.

682 Minkai Xu, Alexander S Powers, Ron O Dror, Stefano Ermon, and Jure Leskovec. Geometric latent
683 diffusion models for 3d molecule generation. In *International Conference on Machine Learning*,
684 pp. 38329–38344, 2023.

685 Ge Yan, Huaijin Wu, and Junchi Yan. Quantum 3d graph learning with applications to molecule
686 embedding. In *International Conference on Machine Learning*, pp. 39126–39137, 2023.

687 Xinyu Ye, Ge Yan, and Junchi Yan. Vqne: Variational quantum network embedding with application
688 to network alignment. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge*
689 *Discovery and Data Mining*, pp. 3105–3115, 2023.

690 Lingxiao Zhao, Xueying Ding, and Leman Akoglu. Pard: Permutation-invariant autoregressive
691 diffusion for graph generation. *Advances in Neural Information Processing Systems*, 37:7156–
692 7184, 2024.

693 Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang,
694 Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applica-
695 tions. *AI open*, 1:57–81, 2020.

702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755

APPENDIX

Table of Contents

A Quantum Basics	15
B Algorithms	15
C Quantum Computing Details	17
C.1 Derivation of the Unified Normalization Term	17
C.2 Quantum Ansatz Design	17
C.3 Training-Free Downsampling and Upsampling Operators	18
D Loss Function Details	18
E Conditional Generation Details	19
F Sensitivity Analysis Details	20
G Experiment Configuration Details	20
H Molecule Visualizations	21
I Future Work and Limitations	21
J The Use of Large Language Models	22

756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809

A QUANTUM BASICS

Quantum computing is a computational paradigm based on the principles of quantum mechanics (Nielsen & Chuang, 2010). Unlike classical bits, which can only be in a state of 0 or 1, the fundamental unit of quantum information is the **qubit**.

A qubit can exist in a state of $|0\rangle$, $|1\rangle$, or a linear combination of both, a property known as **superposition**. The state of a single qubit, $|\psi\rangle$, can be written as:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \tag{14}$$

where α and β are complex numbers called probability amplitudes, satisfying the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. The values $|\alpha|^2$ and $|\beta|^2$ represent the probabilities of the qubit collapsing to the state $|0\rangle$ or $|1\rangle$, respectively, upon measurement. A system of n qubits can exist in a superposition of all 2^n possible classical states, providing access to an exponentially large computational space.

Quantum computations are performed by applying **quantum gates**, which are unitary transformations that evolve the quantum state. A unitary matrix U has the property that its conjugate transpose U^\dagger is also its inverse, i.e., $U^\dagger U = I$. This property ensures that quantum evolution is reversible and conserves probability. Common single-qubit gates include the Hadamard gate (H) for creating superpositions and Pauli gates (X , Y , Z). Multi-qubit gates, such as the Controlled-NOT (CNOT) gate, create **entanglement**, a quantum correlation where the state of one qubit is intrinsically linked to another, regardless of the distance separating them (Nielsen & Chuang, 2010).

In hybrid quantum-classical algorithms, a **VQC** is used, comprising a parameterized **ansatz** $U(\theta)$. The ansatz is a circuit template with tunable parameters θ , optimized classically to minimize a cost function and approximate target states or solutions (Nielsen & Chuang, 2010).

Finally, to extract information from a quantum system, a **measurement** is performed. In the standard computational basis ($\{|0\rangle, |1\rangle\}$), measuring the qubit $|\psi\rangle$ causes its superposition to collapse to one of the basis states. The outcome is probabilistic, governed by the amplitudes of the state vector. For a general n -qubit state $|\Psi\rangle = \sum_{i=0}^{2^n-1} c_i |i\rangle$, a measurement will yield the outcome corresponding to the basis state $|i\rangle$ with probability $|c_i|^2$.

B ALGORITHMS

In this section, we provide the pseudocode for the sampling procedures of our SQ-Diff model. Algorithm 2 details the procedure for random molecular generation using the DDIM sampler. Algorithm 3 extends this to conditional generation using a classifier-guidance strategy.

810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863

Algorithm 2 Random Generation

1: **Input:** Trained parameters θ , number of sampling steps S , schedule parameter η .
2: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ \triangleright Sample initial noise from a standard normal distribution
3: $\mathbf{x}_T \leftarrow \mathbf{x}_T / \|\mathbf{x}_T\|_2$
4: Define timestep sequence $\tau_S, \tau_{S-1}, \dots, \tau_1$ (from T down to 1).
5: **for** $i = S, \dots, 1$ **do**
6: $t \leftarrow \tau_i, t_{\text{prev}} \leftarrow \tau_{i-1}$ (where $\tau_0 = 0$)
7: $|x_t\rangle \leftarrow \text{PrepareStateFromVector}(\mathbf{x}_t, t)$
8: $|\hat{\mathbf{x}}_0\rangle \leftarrow \text{QuantumUNet}(|x_t\rangle, t; \theta)$ \triangleright Predict the original state vector
9: $\hat{\mathbf{x}}_0 \leftarrow \text{GetAmplitudes}(|\hat{\mathbf{x}}_0\rangle)$
10: $\hat{\boldsymbol{\epsilon}}_\theta \leftarrow (\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \hat{\mathbf{x}}_0) / \sqrt{1 - \bar{\alpha}_t}$ \triangleright Calculate the corresponding noise
11: $\sigma_t \leftarrow \eta \sqrt{\frac{1 - \bar{\alpha}_{t_{\text{prev}}}}{1 - \bar{\alpha}_t}} \sqrt{1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t_{\text{prev}}}}}$ \triangleright Noise schedule
12: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ \triangleright Sample fresh noise per step
13: $\mathbf{x}_{t_{\text{prev}}} \leftarrow \sqrt{\bar{\alpha}_{t_{\text{prev}}}} \hat{\mathbf{x}}_0 + \sqrt{1 - \bar{\alpha}_{t_{\text{prev}}} - \sigma_t^2} \cdot \hat{\boldsymbol{\epsilon}}_\theta + \sigma_t \boldsymbol{\epsilon}$ \triangleright DDIM update step
14: $\mathbf{x}_{t_{\text{prev}}} \leftarrow \mathbf{x}_{t_{\text{prev}}} / \|\mathbf{x}_{t_{\text{prev}}}\|_2$ \triangleright Normalize vector for next step
15: **end for**
16: $M \leftarrow \text{ReverseReconstruction}(\mathbf{x}_0)$ \triangleright Reconstruct molecule from final vector
17: **return** M

Algorithm 3 Conditional Generation

1: **Input:** Trained parameters θ , predictor $r_\phi(\cdot)$, target property y , guidance strength λ , steps S , schedule η .
2: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and normalize \mathbf{x}_T .
3: Define timestep sequence $\tau_S, \tau_{S-1}, \dots, \tau_1$.
4: **for** $i = S, \dots, 1$ **do**
5: $t \leftarrow \tau_i, t_{\text{prev}} \leftarrow \tau_{i-1}$
6: $|x_t\rangle \leftarrow \text{PrepareStateFromVector}(\mathbf{x}_t, t)$
7: $|\hat{\mathbf{x}}_{0, \text{uncond}}\rangle \leftarrow \text{QuantumUNet}(|x_t\rangle, t; \theta)$
8: $\hat{\mathbf{x}}_{0, \text{uncond}} \leftarrow \text{GetAmplitudes}(|\hat{\mathbf{x}}_{0, \text{uncond}}\rangle)$
9: \triangleright Compute guidance gradient
10: With `torch.enable_grad()`:
11: $\mathbf{x}_{\text{grad}} \leftarrow \hat{\mathbf{x}}_{0, \text{uncond}} \cdot \text{detach}().\text{requires_grad}()$
12: $\mathcal{L}_{\text{prop}} \leftarrow \text{MSE}(r_\phi(\mathbf{x}_{\text{grad}}), y)$
13: $\mathbf{g} \leftarrow \nabla_{\mathbf{x}_{\text{grad}}} \mathcal{L}_{\text{prop}}$
14: $\hat{\mathbf{x}}_0 \leftarrow \hat{\mathbf{x}}_{0, \text{uncond}} - \lambda \cdot \mathbf{g}$ \triangleright Apply guidance to the prediction
15: $\hat{\mathbf{x}}_0 \leftarrow \hat{\mathbf{x}}_0 / \|\hat{\mathbf{x}}_0\|_2$ \triangleright Perform DDIM update using the guided $\hat{\mathbf{x}}_0$
16: $\hat{\boldsymbol{\epsilon}}_\theta \leftarrow (\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \hat{\mathbf{x}}_0) / \sqrt{1 - \bar{\alpha}_t}$
17: $\sigma_t \leftarrow \eta \sqrt{\frac{1 - \bar{\alpha}_{t_{\text{prev}}}}{1 - \bar{\alpha}_t}} \sqrt{1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t_{\text{prev}}}}}$
18: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
19: $\mathbf{x}_{t_{\text{prev}}} \leftarrow \sqrt{\bar{\alpha}_{t_{\text{prev}}}} \hat{\mathbf{x}}_0 + \sqrt{1 - \bar{\alpha}_{t_{\text{prev}}} - \sigma_t^2} \cdot \hat{\boldsymbol{\epsilon}}_\theta + \sigma_t \boldsymbol{\epsilon}$
20: $\mathbf{x}_{t_{\text{prev}}} \leftarrow \mathbf{x}_{t_{\text{prev}}} / \|\mathbf{x}_{t_{\text{prev}}}\|_2$
21: **end for**
22: $M \leftarrow \text{ReverseReconstruction}(\mathbf{x}_0)$
23: **return** M

C QUANTUM COMPUTING DETAILS

C.1 DERIVATION OF THE UNIFIED NORMALIZATION TERM

We construct a molecular structure vector \mathbf{m} such that its squared L2-norm is independent of the specific molecular geometry, depending only on the number of atoms N . This is a crucial step for the amplitude encoding of the initial quantum state (Smaldone et al., 2025). To achieve this, we introduce a specific correction term:

$$v_{\text{corr}} = \frac{\sqrt{2} \cdot N \|\mathbf{a}_{\min}\|_2}{\max_{1 \leq i, j \leq N} \|\mathbf{a}_i - \mathbf{a}_j\|_\infty}, \quad (15)$$

The squared L2-norm of the flattened feature vector \mathbf{m} is then calculated as follows:

$$\begin{aligned} \|\mathbf{m}\|_2^2 &= \underbrace{\sum_{i=1}^N \|\mathbf{a}_i\|_2^2}_{\text{coordinates}} + \underbrace{\sum_{i=1}^N 1}_{\text{atom types}} + \underbrace{(2N+1) \sum_{i=1}^N (3 - \|\mathbf{a}_i\|_2^2)}_{\text{auxiliary values}} + \underbrace{\sum_{i,j=1}^N \|\mathbf{a}_i - \mathbf{a}_j\|_2^2}_{\text{distances}} + \underbrace{v_{\text{corr}}^2}_{\text{correction}} \\ &= 6N^2 + 4N + \sum_{i=1}^N \|\mathbf{a}_i\|_2^2 + 2N \sum_{i=1}^N \|\mathbf{a}_i\|_2^2 - 2 \sum_{\text{cyc}} \left(\sum_{i=1}^N a_{i\bullet} \right)^2 - (2N+1) \sum_{i=1}^N \|\mathbf{a}_i\|_2^2 \\ &\quad + v_{\text{corr}}^2 \\ &= 6N^2 + 4N + (1 + 2N - (2N+1)) \sum_{i=1}^N \|\mathbf{a}_i\|_2^2 - 2 \sum_{\text{cyc}} \left(\sum_{i=1}^N \frac{a_{i\bullet}^{\text{ori}} - a_{\min,\bullet}}{D} \right)^2 + v_{\text{corr}}^2 \\ &= 6N^2 + 4N - \frac{2N^2 \|\mathbf{a}_{\min}\|_2^2}{D^2} + v_{\text{corr}}^2 \\ &= 6N^2 + 4N, \end{aligned} \quad (16)$$

where $(a_{ix}^{\text{ori}}, a_{iy}^{\text{ori}}, a_{iz}^{\text{ori}})$ are the original coordinates of the i -th atom before translation, $\mathbf{a}_{\min} = (\min_N a_{ix}^{\text{ori}}, \min_N a_{iy}^{\text{ori}}, \min_N a_{iz}^{\text{ori}})$, and $D = \max_{1 \leq i, j \leq N} \|\mathbf{a}_i - \mathbf{a}_j\|_\infty$. Here, $\sum_{\text{cyc}}(\cdot)$ denotes a cyclic sum over the three coordinate axes x, y, z (the bullet \bullet indicates the axis being summed). This simplification provides a unified normalization term $\sqrt{6N^2 + 4N}$ for all molecules with N atoms. To ensure the feature dimension is exactly 2^q for amplitude encoding, we pad the remaining dimensions with 0. The initial quantum state $|x_0\rangle$ is prepared as follows:

$$\begin{aligned} |x_0\rangle &= \frac{1}{\sqrt{6N^2 + 4N}} \sum_{i=1}^N \left(a_{ix} |r_i\rangle + a_{iy} |r_i + 1\rangle + a_{iz} |r_i + 2\rangle + \sum_{j=1}^M \mathbf{1}_{ij} |r_i + 2 + j\rangle \right) \\ &\quad + \sum_{k=1, k \neq i}^N \|\mathbf{a}_i - \mathbf{a}_k\|_2 |r_i + M + 2 + k\rangle + \sqrt{(2N+1)(3 - \|\mathbf{a}_i\|_2^2)} |r_i + M + N + 3\rangle \\ &\quad + \frac{v_{\text{corr}}}{\sqrt{6N^2 + 4N}} |r_{N+1}\rangle + \sum_{i=N(M+N+3)+2}^{2^q} 0 |r_{N+2} + i\rangle, \end{aligned} \quad (17)$$

where M is the number of atom types, q is the number of qubits, and r_i is the index of the i -th block.

C.2 QUANTUM ANSATZ DESIGN

Each learnable block in our Quantum U-Net is a composite of VQCs consisting of ansatzes, as depicted in Figure 3. The design of the ansatz is crucial for the model’s expressive power (Cerezo et al., 2021). Our VQC layer is composed of two primary components: a layer of single-qubit rotations and a layer of entangling gates.

The single-qubit rotation layer applies a sequence of three rotations, R_z , R_y , and another R_z , to each qubit independently. This structure is a universal single-qubit gate, meaning it can create any arbitrary single-qubit state (Barenco et al., 1995; Nielsen & Chuang, 2010). For an n -qubit system, this layer can be expressed as:

$$\mathbf{U}_{\text{rot}}(\boldsymbol{\theta}_{\text{rot}}) = \bigotimes_{i=0}^{n-1} R_z(\theta_{i,2})R_y(\theta_{i,1})R_z(\theta_{i,0}) \quad (18)$$

Following the rotation layer, an entangling layer is applied to create correlations between the qubits. We employ a linear entanglement scheme using Controlled- R_x (CR_x) gates. A $CR_x(\theta)$ gate applies an $R_x(\theta)$ rotation to a target qubit conditional on the state of a control qubit. This is performed sequentially for adjacent pairs of qubits, a common structure in hardware-efficient ansatzes (Cerezo et al., 2021):

$$\mathbf{U}_{\text{ent}}(\boldsymbol{\theta}_{\text{ent}}) = \prod_{i=0}^{n-2} CR_x^{(i,i+1)}(\theta_{i,3}) \quad (19)$$

where $CR_x^{(i,j)}$ denotes a CR_x gate with control qubit i and target qubit j . The full VQC layer is the product of these two components, $\mathbf{U}(\boldsymbol{\theta}) = \mathbf{U}_{\text{ent}}(\boldsymbol{\theta}_{\text{ent}})\mathbf{U}_{\text{rot}}(\boldsymbol{\theta}_{\text{rot}})$, where $\boldsymbol{\theta}$ comprises all trainable rotation angles.

C.3 TRAINING-FREE DOWNSAMPLING AND UPSAMPLING OPERATORS

A core design principle of our framework is to maintain a pure quantum architecture. To this end, we designed downsampling and upsampling operators that are parameter-free and interleave naturally with the unitary operations of the quantum ansatzes.

Downsampling Operator. Our downsampling operator is inspired by the partial trace, a technique used in quantum autoencoders to reduce dimensionality (Romero et al., 2017). To downsample an n -qubit system to an m -qubit system ($m < n$), we partition the Hilbert space into $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$, where \mathcal{H}_A has dimension 2^m and \mathcal{H}_B has dimension 2^{n-m} . The state can be written as $|\psi\rangle = \sum_{i=0}^{2^m-1} \sum_{j=0}^{2^{n-m}-1} c_{ij} |i\rangle_A |j\rangle_B$. Our operator projects the ancillary subsystem B onto the basis state $|0\rangle_B$. The resulting unnormalized state of subsystem A is $|\tilde{\psi}_A\rangle = \sum_{i=0}^{2^m-1} c_{i0} |i\rangle_A$. The final downsampled pure state is then obtained by normalization:

$$|\psi_A\rangle = \frac{1}{\sqrt{\sum_{i=0}^{2^m-1} |c_{i0}|^2}} \sum_{i=0}^{2^m-1} c_{i0} |i\rangle_A \quad (20)$$

This procedure effectively reduces the Hilbert space dimensionality while preserving a pure state representation for the subsequent quantum ansatzes.

Upsampling Operator. For the upsampling path, we employ a linear interpolation technique, analogous to transpose convolutions in classical U-Nets (Ronneberger et al., 2015). Given an n -qubit state $|\psi\rangle_n = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle_n$, we produce an $(n+1)$ -qubit state $|\phi\rangle_{n+1}$. We compute an unnormalized amplitude vector $\tilde{\beta}$ of size 2^{n+1} by interpolating between adjacent amplitudes in α :

$$\tilde{\beta}_{2i} = w_0 \alpha_i + w_1 \alpha_{i+1} \quad (21)$$

$$\tilde{\beta}_{2i+1} = w_1 \alpha_i + w_0 \alpha_{i+1} \quad (22)$$

for $i = 0, \dots, 2^n - 2$ where w_0 and w_1 are fixed, non-learnable hyperparameters. This linear transformation can be expressed via a sparse matrix operator \mathcal{L} acting on the amplitude vector. The final state is normalized to have a unit norm:

$$|\phi\rangle_{n+1} = \frac{1}{\|\tilde{\beta}\|_2} \sum_{j=0}^{2^{n+1}-1} \tilde{\beta}_j |j\rangle_{n+1} \quad (23)$$

D LOSS FUNCTION DETAILS

The total loss function is $\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{prior}} \mathcal{L}_{\text{prior}} + \lambda_{\text{perm}} \mathcal{L}_{\text{perm}}$.

972 **Reconstruction Loss.** We use an ℓ_2 loss on the clean-state prediction:

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{\mathbf{x}_0, \epsilon, t} [\|\mathbf{x}_0 - \hat{\mathbf{x}}_0(\mathbf{x}_t, t)\|_2^2] \quad (24)$$

973 where $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon$ and $\hat{\mathbf{x}}_0(\cdot)$ denotes the Quantum U-Net prediction of the clean state.

974 **Classification Loss.** We use Focal Loss (Lin et al., 2017) to improve atom type prediction. Let \hat{y}_i be the predicted probability vector for the type of atom i , extracted from the denoised state $\hat{\mathbf{x}}_0$. The loss is:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^M y_{ic} (1 - \hat{y}_{ic})^\gamma \log(\hat{y}_{ic}) \quad (25)$$

975 where y_{ic} is 1 if atom i is of type c and 0 otherwise, and γ is a focusing parameter.

976 **Graph Prior Loss.** This loss enforces internal consistency. It has two parts, $\mathcal{L}_{\text{prior}} = \mathcal{L}_{\text{onehot}} + \mathcal{L}_{\text{dist}}$. The one-hot loss ensures that predicted atom type distributions are valid:

$$\mathcal{L}_{\text{onehot}} = \frac{1}{N} \sum_{i=1}^N \left(\sum_{c=1}^M \hat{y}_{ic} - 1 \right)^2 \quad (26)$$

977 The distance consistency loss enforces geometric agreement between reconstructed coordinates $\hat{\mathbf{a}}_i$ and reconstructed distances \hat{d}_{ij} , a common practice in 3D generative models (Hoogeboom et al., 2022):

$$\mathcal{L}_{\text{dist}} = \frac{1}{N(N-1)} \sum_{i \neq j} \left(\|\hat{\mathbf{a}}_i - \hat{\mathbf{a}}_j\|_2 - \hat{d}_{ij} \right)^2 \quad (27)$$

978 **Permutation Invariance Loss.** To enforce permutation invariance, we sample a random permutation π and penalize differences between the permuted output and the output of the permuted input. This is a key inductive bias for graph-structured data (Zhou et al., 2020). Let $\hat{\mathbf{x}}_0 = f_\theta(\mathbf{x}_t)$ be the denoising function.

$$\mathcal{L}_{\text{perm}} = \mathbb{E}_{\mathbf{x}_t, \pi} [\|\pi(f_\theta(\mathbf{x}_t)) - f_\theta(\pi(\mathbf{x}_t))\|_2^2] \quad (28)$$

979 where $\pi(\cdot)$ is an operator that permutes the atom-related entries in the state vector.

1004 E CONDITIONAL GENERATION DETAILS

1005 This section provides details on the chemical properties used for conditional generation, as discussed in Section 4.3. The guidance mechanism relies on a pre-trained classical classifier for each property. We constructed a simple MLP for each of the six properties. The input to each MLP is the flattened molecular feature vector prepared for amplitude encoding, as described in Section 3.1. The six properties are defined as follows:

1006 **Polarizability** (α , in Bohr³): A measure of how easily the electron cloud of a molecule can be distorted by an external electric field.

1007 **HOMO Energy** (ϵ_{HOMO} , in Hartree): The energy of the highest occupied molecular orbital, related to the molecule’s capacity to donate an electron.

1008 **LUMO Energy** (ϵ_{LUMO} , in Hartree): The energy of the lowest unoccupied molecular orbital, related to the molecule’s capacity to accept an electron.

1009 **HOMO-LUMO Gap** ($\Delta\epsilon$, in Hartree): The energy difference between HOMO and LUMO, which is an indicator of molecular stability.

1010 **Dipole Moment** (μ , in Debye): A measure of the net molecular polarity, resulting from the separation of positive and negative charges.

1011 **Heat Capacity** (C_v , in cal/(mol·K)): The amount of heat required to raise the temperature of one mole of the substance by one degree Kelvin at a constant volume.

1012 Table 3 provides the detailed numerical results for the conditional generation experiments.

Table 3: Comparison of property distributions for 10K molecules generated by SQ-Diff. For the random (baseline) model, $Q_{0.25}$ and $Q_{0.75}$ denote the quartiles of the generated distribution. For the guided model, the columns show the medians of the generated distributions conditioned on the lower ($Q_{0.25}$) and upper ($Q_{0.75}$) quartiles of the target distribution from QM9. RE quantifies the relative error $|\text{median} - \text{target } Q|/|\text{target } Q|$ for each guided condition.

Property	SQ-Diff (random)		SQ-Diff (guided)		RE	
	$Q_{0.25}$	$Q_{0.75}$	Median ($Q_{0.25}$)	Median ($Q_{0.75}$)	$Q_{0.25}$	$Q_{0.75}$
α	61.13	62.84	61.39	62.59	0.419%	0.409%
$\Delta\varepsilon$	0.2774	0.2910	0.2782	0.2899	0.282%	0.406%
$\varepsilon_{\text{HOMO}}$	-0.2578	-0.2506	-0.2578	-0.2507	0.002%	0.044%
$\varepsilon_{\text{LUMO}}$	0.0284	0.0385	0.0293	0.0378	3.244%	1.783%
μ	3.285	3.346	3.290	3.339	0.140%	0.222%
C_v	35.79	37.55	36.04	37.29	0.714%	0.689%

F SENSITIVITY ANALYSIS DETAILS

Table 4 provides the data for the sensitivity analysis of the DDIM noise ratio hyperparameter (η), as discussed in Section 4.4.

Table 4: Sensitivity analysis data for DDIM noise ratio η .

η	V (%)	V×U (%)	V×N (%)	V×U×N (%)	T_{eff} (s)
0.00	81.1	46.2	90.0	46.1	0.130
0.01	80.1	48.3	79.9	48.1	0.125
0.02	75.9	52.4	75.7	52.3	0.115
0.03	69.8	55.1	69.6	55.0	0.109
0.04	62.7	54.5	62.4	54.3	0.111
0.05	55.1	50.3	54.7	50.0	0.120

G EXPERIMENT CONFIGURATION DETAILS

Table 5: Hardware configuration and model hyperparameters used for all experiments.

Hardware Configuration	
GPU	NVIDIA GeForce RTX 3090
CPU	Intel Xeon E5-2683 v4
Memory	128 GB
Model Hyperparameters	
Batch Size	512
Number of Epochs	50
Learning Rate	1×10^{-4}
Weight Decay	1×10^{-4}
Number of Qubits	8
Number of VQC Blocks	5
DDIM Sampling Steps	10
DDIM η	0.01

All experiments were conducted using the hardware and hyperparameter settings detailed in Table 5. The selection of hyperparameters was guided by common practices in related QML literature and grid search. Our model is trained using the TorchQuantum (Wang et al., 2022) simulator, which represents VQCs as a series of unitary operations. These operations are executed as standard tensor manipulations within the PyTorch framework. This design choice ensures that the evolution of the quantum state vector is fully differentiable. Consequently, gradients with respect to the VQC parameters can be computed directly via backpropagation, enabling end-to-end training of the model using standard deep learning optimization techniques.

Regarding the coefficients for the loss components (λ_{cls} , λ_{prior} , λ_{perm}) detailed in Section 3.3.3, it is important to note that these are not treated as conventional hyperparameters for tuning. Instead, they serve as static scaling factors. Their primary purpose is to balance the magnitudes of the gradients originating from different loss terms, ensuring that they are on a comparable scale during backpropagation.

H MOLECULE VISUALIZATIONS

We provide additional qualitative examples of molecules generated by SQ-Diff; see Figure 6. Note that hydrogen atoms are not explicitly modeled in our method and are added post-hoc using RD-Kit (Landrum et al., 2013) toolkit for visualization.

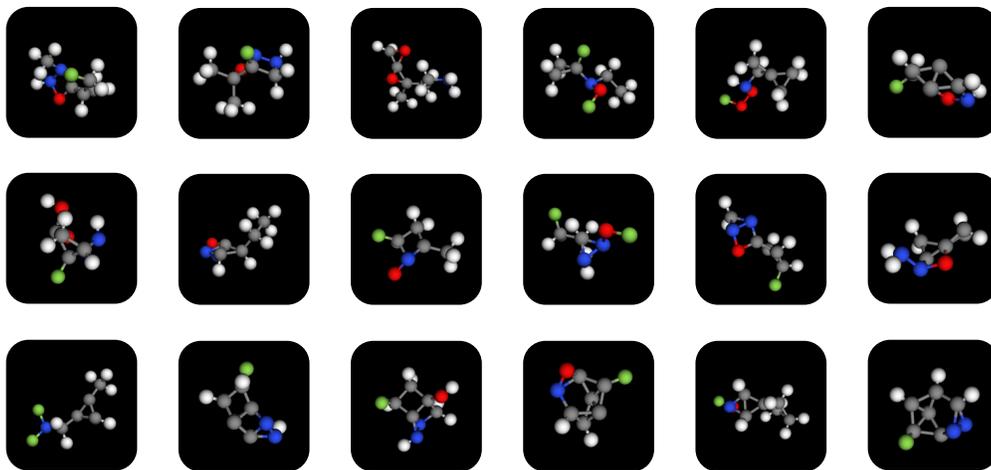


Figure 6: Examples of 3D molecules randomly generated by SQ-Diff.

I FUTURE WORK AND LIMITATIONS

Our work represents a significant step towards practical quantum generative modeling for scientific applications, but several limitations and avenues for future research remain. The primary limitation of this study is its reliance on classical simulators for VQCs. While our model is designed with NISQ hardware in mind, its performance on real quantum devices remains to be demonstrated and will be subject to hardware noise and qubit connectivity constraints (Bharti et al., 2022). An immediate and crucial next step is the deployment and validation of SQ-Diff on actual quantum hardware. This will provide invaluable insights into the practical feasibility and potential quantum advantage of our approach. A second limitation is the performance gap in generation quality compared to leading classical models. From a modeling perspective, exploring quantum analogues of more advanced classical generative frameworks, such as flow-matching models (Hong et al., 2025), could lead to further improvements in generation quality and efficiency. Additionally, extending the framework to handle larger and more complex molecular systems, such as proteins, presents an exciting and challenging direction for future work.

1134 J THE USE OF LARGE LANGUAGE MODELS

1135

1136 We acknowledge the use of a large language model (LLM) as a writing assistant in the preparation
1137 of this manuscript. The LLM was utilized for two primary purposes: (1) to polish the language and
1138 improve the clarity, conciseness, and overall expression of the text; and (2) to help identify potential
1139 gaps or areas for improvement in the narrative’s logical flow. We emphasize that the LLM’s role was
1140 strictly that of a writing and editing tool. The core scientific ideas, the proposed methodology, the
1141 experimental results, and their interpretation are entirely the work of the authors.

1142

1143

1144

1145

1146

1147

1148

1149

1150

1151

1152

1153

1154

1155

1156

1157

1158

1159

1160

1161

1162

1163

1164

1165

1166

1167

1168

1169

1170

1171

1172

1173

1174

1175

1176

1177

1178

1179

1180

1181

1182

1183

1184

1185

1186

1187