Crystal Structure Generation Using a Diffusion Model Conditioned on X-Ray Diffraction Intensities Without Label Learning

Kenta Tsukaue [1], Koji Yasuda [1,2]

[1]Graduate School of Informatics, Nagoya University
[2]Institute of Materials and Systems for Sustainability, Nagoya University

tsukaue.kenta.e6@s.mail.nagoya-u.ac.jp, yasudak@imass.nagoya-u.ac.jp

Abstract

We propose a method to apply diffusion models, commonly used for image generation, to generate crystal structures by conditioning on X-ray diffraction intensities. Traditionally, image generation is conditioned using label-based learning and prompts, but there is growing interest in conditioning based on the Tweedy formula, which does not require additional training. On the other hand, traditional X-ray crystallography use the physical model of X-ray diffraction together with chemical knowledge to solve structure. From the theory of X-ray diffraction, we derived an expression for the conditional score of the diffusion model. The method is implemented in a pretrained diffusion model, and its property is examined. This research develops a method to utilize powerful empirical priors to a scientific inference without using label-based learning.

Introduction

Diffusion models have gained significant attention in the field of image generation, and their applications have recently expanded to the field of chemistry, including structure generation of molecules, proteins, and crystals. In crystal structure generation, CDVAE (Xie et al. 2021) showed that with suitable neural networks, the variational autoencoder (VAE) can generate crystal structures. Later, DiffCSP (Jiao et al. 2023) emerged as an accurate generative model solely based on the diffusion model. It enables the generation of structures similar to those in the training data by specifying the types and numbers of atoms in a crystal.

These studies unlock new possibilities for inverse problems as they allow for the incorporation of strong empirical priors into the process of scientific inference. Unlike image generation, however, this field is not mature enough to generate specific structures with prompts. The present study aims to utilize X-ray diffraction intensities as prompts for crystal structure generation. Traditional conditional generation methods, including classifier guidance (Dhaliwal and Nichol 2021) and classifier-free guidance (Ho and Salimans 2021), require a large number of sample-prompt pairs, leading to high computational costs. However, as the physics of the X-ray diffraction is completely known, it is possible to derive analytically the conditional score, or the guidance. Using the torus-based diffusion process in crystal structure generation and the Tweedy formula on it, we demonstrated this and implemented it in the pretrained diffusion model. We applied this method to several systems and showed that structures that reproduce diffraction intensities are preferentially generated. While the method struggles with structures containing mixed heavy and light atoms or a large number of atoms, as sufficient guidance cannot be provided during generation, its effectiveness has been demonstrated for data with fewer atoms and without such mixtures.

Related Works

Definitions

A crystal $\mathcal{M}$ is specified with the unit cell formula $A \in R^{h\times N}$, crystal lattice vector $L \in R^{3\times 3}$, and the atomic coordinates within the unit cell $X \in R^{3\times N}$, where N is the number of atoms in the unit cell. Crystals have symmetry; the translation or the rotation as a whole does not change a crystal, and it has an obvious periodicity. There are multiple $\mathcal{M}$s that represent the same crystal, yet the crystal generation model should yield the same result no matter which one is used. This enforces on the model the stringent conditions, the permutation and periodic invariance, and the SE(3) equivariance, which means that inputting a rotated (translated) structure to a model will output a vector rotated (translated) by the same amount. Many graph neural networks (GNNs) could satisfy these requirements. The characteristics of the data compared to other modalities are as follows. Since the typical N is less than 10^2, the number of dimensions of L and X is small compared to images. The degrees of freedom of A are very large in principle, but much less in the actual elements used. Since the observed 3D structure is

stable, the score should be close to the force on each atom, and the GNN for approximating the molecular force field would be suitable for the score model.

CDVAE

To reconstruct the material from the latent representation, CDVAE first creates an approximate crystal structure with VAE and then improves the structure using score. CDVAE consists of three networks: a GNN that encodes $\mathcal{M}$ into a latent representation $\boldsymbol{z}$, a MLP that approximates crystal structure $\mathcal{M} = (A, L, X)$ from $\boldsymbol{z}$, and a GNN to improve the structure (A, X). The model satisfies permutational and SE(3) equivariance. Periodicity is recovered using several adjacent unit cells. The multi-graph representation to predict bonds between atoms (or to define "neighbor" atoms) is built with CrystalNN (Pan et al. 2021) for the encoder, and with the K-nearest neighbor algorithm (K = 20) for the decoder. The DimeNet++ (Gasteiger et al. 2020) and the successor (GemNet-dQ, Gasteiger et al. 2021) were used as GNNs to incorporate the bending and dihedral angle dependence.

DiffCSP

DiffCSP (Jiao et al. 2023) generates the crystal lattice vectors L and the atomic coordinates (fractional coordinates) X using the Denoising Diffusion Probabilistic Model (DDPM) (Ho, Jain, and Abbeel 2020). The diffusion process of the lattice vector is the standard Ornstein-Uhlenbeck process in $R^{3\times3}$. As in image generation, L_t is reduced toward zero and noise is added to it to learn the score. The transition probability distribution from time 0 to t is the multivariate Normal:

$$q(L_t|L_0) = \mathcal{N}(L_t;\ \sqrt{\bar{\alpha}_t}L_0, (1-\bar{\alpha}_t)I) \tag{1}$$

Here, the noise strength is controlled with the cosine scheduler: $\bar{\alpha}_t = \prod_{s=1}^{t}\alpha_t = \prod_{s=1}^{t}(1-\beta_s) = \cos^2(\pi t/2T)$. In the reverse diffusion process noise is gradually removed, and the transition probability is represented as usual:

$$p(L_{t-1}|\mathcal{M}_t) = \mathcal{N}(L_{t-1};\ m(\mathcal{M}_t), c(\mathcal{M}_t)I) \tag{2}$$

$$m(\mathcal{M}_t) = \frac{1}{\sqrt{\alpha_t}}\left(L_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\hat{\epsilon}_L(\mathcal{M}_t, t)\right) \tag{3}$$

$$c(\mathcal{M}_t) = \beta_t\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \tag{4}$$

The noise removal term $\hat{\epsilon}_L(\mathcal{M}_t, t) \in R^{3\times3}$ is predicted by a neural network model. To train it standard score matching (Song and Ermon 2020; Song et al. 2020) is used: we first sample $\epsilon_L \sim \mathcal{N}(0, I)$, t is sampled from uniform random number, $t \sim U(1, T)$ and obtain a noisy sample $L_t = \sqrt{\bar{\alpha}_t}L_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_L$. The objective function for training is the L2 loss between ϵ_L and $\hat{\epsilon}_L$:

$$\mathcal{L}_L = E_{\substack{\epsilon_L\sim\mathcal{N}(0,I)\\ t\sim U(1,T)}}\left[|\epsilon_L - \hat{\epsilon}_L(\mathcal{M}_t, t)|^2\right] \tag{5}$$

Due to the periodicity of crystals, the diffusion processes for the atomic coordinates needs special care. The domain of the fractional coordinate X, $[0,1)^{3\times N}$, forms a quotient space $R^{3\times N}/Z^{3\times N}$ due to the periodicity. Therefore, we consider the diffusion on the torus without drift term: we sample noise $\epsilon_X \in R^{\mathbb{3}\times N}$, do the random walk from X_0 to $X_0 + \sigma_t\epsilon_X$, then extract fractional coordinate, $X_t = w(X_0 + \sigma_t\epsilon_X)$, where $w(X) = X - \lfloor X \rfloor \in [0,1)^{3\times N}$. The transition probability of this process is given by the Wrapped Normal (WN) distribution (Bortoli et al. 2022).

$$q(X_t|X_0) \propto \sum_{Z\in Z^{3\times N}} \exp\left(-\frac{|X_t - X_0 + Z|^2}{2\sigma_t^2}\right) \tag{6}$$

The noise scale σ_t obeys the exponential scheduler.

$$\frac{\sigma_t}{\sigma_1} = \left(\frac{\sigma_T}{\sigma_1}\right)^{\frac{t-1}{T-1}} \tag{7}$$

The WN distribution has been used in molecular structure generation (Jing et al. 2022). The objective function for score matching is:

$$\mathcal{L}_X = E_{\substack{X_t\sim q(X_t|X_0)\\ t\sim U(1,T)}}\left[\lambda_t|\nabla_{X_t}\log q\,(X_t|X_0) - \hat{\epsilon}_X(\mathcal{M}_t, t)|^2\right] \tag{8}$$

Here, λ_t^{-1} is the averaged 2-norm of the score,

$$\lambda_t^{-1} = E_{X_t\sim q(X_t|0)}\left[\nabla_{X_t}\log q\,(X_t|0)|^2\right] \tag{9}$$

To approximate score DiffCSP uses special EGNN that satisfies the periodicity. Denoting latent representation of the v-th atom as $\boldsymbol{h}_v$, the message from the ξ-th atom to the v-th atom is

$$m_{v\xi} = \mathrm{MLP}\left(\boldsymbol{h}_v, \boldsymbol{h}_\xi, L^T L, \Psi_{FT}(\boldsymbol{x}_v - \boldsymbol{x}_\xi)\right) \tag{10}$$

$$\Psi_{FT}(\boldsymbol{x}_v - \boldsymbol{x}_\xi) = \left(\sin 2\pi m(\boldsymbol{x}_v - \boldsymbol{x}_\xi), \cos 2\pi m(\boldsymbol{x}_v - \boldsymbol{x}_\xi)\right) \tag{11}$$

Relative coordinate $\boldsymbol{x}_v - \boldsymbol{x}_\xi$ is defined by the three real numbers, but Ψ_{FT} expresses it with the 256 point values of the 6 periodic functions of m.

After updating the latent representations for 4-6 times,

$$\boldsymbol{h}_v \to \boldsymbol{h}_v + \mathrm{MLP}\left(\boldsymbol{h}_v, \sum_{\xi=1}^{N} m_{v\xi}\right) \tag{12}$$

they are used to approximate the score.

$$\hat{\epsilon}_L = L\cdot\mathrm{MLP}\left(\frac{1}{N}\sum_{v=1}^{N}\boldsymbol{h}_v\right) \tag{13}$$

$$\hat{\epsilon}_{\boldsymbol{x}_v} = \mathrm{MLP}(\boldsymbol{h}_v) \tag{14}$$

Classifier/Classifier-Free Guidance

To generate $\boldsymbol{x}$ under condition $\boldsymbol{y}$, we need the conditional distribution and its score. Using Bayes' theorem

$$p_{t|y}(\boldsymbol{x}_t|\boldsymbol{y}) = \frac{p_{y|t}(\boldsymbol{y}|\boldsymbol{x}_t)p_t(\boldsymbol{x}_t)}{p(\boldsymbol{y})} \quad (15)$$

we obtain the following equation.

$$\nabla_{\boldsymbol{x}_t} \log p_{t|y}(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t} \log p_{\boldsymbol{y}|t}(\boldsymbol{y}|\boldsymbol{x}_t) \quad (16)$$

The last term in the right-hand side (RHS) is the score of the classifier $p_\theta(\boldsymbol{y}|\boldsymbol{x_t})$ that predicts the label $\boldsymbol{y}$ on noisy sample. According to (Dhariwal and Nichol 2021) Eqs. (2) and (3) become

$$p(L_{t-1}|\mathcal{M}_t, y) = \mathcal{N}(L_{t-1}; \tilde{m}(\mathcal{M}_t), c(\mathcal{M}_t)I) \quad (17)$$

$$\tilde{m}(\mathcal{M}_t) = m(\mathcal{M}_t) + \omega c(\mathcal{M}_t)\nabla_{x_t} \log p_\theta\,(y|\mathcal{M}_t) \quad (18)$$

The scaling parameter ω adjusts the guidance strength.

Classifier-free guidance is intended to generate samples in a given class $\boldsymbol{y}$ using only score estimator $\epsilon_\theta(\boldsymbol{x}_t, \boldsymbol{y})$, without separate classifier models. Unconditional means a null token ϕ for the class $\boldsymbol{y}$. Sampling is performed using the following linear combination of the scores:

$$\tilde{\epsilon}_\theta(\boldsymbol{x}_t, \boldsymbol{y}) = (1-\omega)\epsilon_\theta(\boldsymbol{x}_t, \boldsymbol{y} = \phi) + \omega\epsilon_\theta(\boldsymbol{x}_t, \boldsymbol{y}) \quad (19)$$

This has been very successful with the image generation model. The neural network for the score should also accept the condition $\boldsymbol{y}$, which poses a particular challenge in crystal generation.

Tweedie Moment Projected Diffusions

Classifier- and classifier-free learning need large amounts of data and retraining for each specific task. If the process of generating label $\boldsymbol{y}$ is explicitly known, Tweedie Moment Projected Diffusions (TMPD) (Boys et al. 2023) gives a way to avoid them. Its purpose is to preferentially generate samples $\boldsymbol{x}_0$ that explain noisy measurements $\boldsymbol{y}$, when H and σ_y are known.

$$\boldsymbol{y} = H\boldsymbol{x}_0 + \boldsymbol{u}, \quad \boldsymbol{u} \sim \mathcal{N}(0, \sigma_y^2 I) \quad (20)$$

The score function under this condition has a correction $\nabla_{\boldsymbol{x}_t} \log p_{\boldsymbol{y}|t}(\boldsymbol{y}|\boldsymbol{x}_t)$ and below we express it in terms of $\boldsymbol{x}_t$ and $\boldsymbol{y}$. Because of Eq. (20), $p(\boldsymbol{y}|\boldsymbol{x}_0)$ is Gaussian, and assuming $p_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t) \approx \mathcal{N}(\boldsymbol{x}_0; \boldsymbol{m}(\boldsymbol{x}_t), C(\boldsymbol{x}_t))$, we have

$$p_{\boldsymbol{y}|t}(\boldsymbol{y}|\boldsymbol{x}_t) = \int p_{\boldsymbol{y}|0}(\boldsymbol{y}|\boldsymbol{x}_0)\, p_{0|t}\,(\boldsymbol{x}_0|\boldsymbol{x}_t) d\boldsymbol{x}_0 = \mathcal{N}(\boldsymbol{y}\,;\, H\boldsymbol{m}, HCH^\top + \sigma_y^2 I) \quad (21)$$

meaning that the correction can be represented with $\boldsymbol{m}$, C, H, and σ_y. Next, given the marginal density $p_t(\boldsymbol{x}_t)$, the mean of $p_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t)$ establishes the relation between $\boldsymbol{m}$ and $\boldsymbol{x}_t$, called the Tweedy formula,

$$\boldsymbol{m} = E[\boldsymbol{x}_0|\boldsymbol{x}_t] = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + v_t \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)\right) \quad (22)$$

where $v_t = 1 - \alpha_t$. Similarly, the covariance yields the formula of C.

$$C = E[(\boldsymbol{x}_0 - \boldsymbol{m})(\boldsymbol{x}_0 - \boldsymbol{m})^\top|\boldsymbol{x}_t] = \frac{v_t}{\alpha_t}(I + v_t \nabla^2 \log p_t(\boldsymbol{x}_t)) = \frac{v_t}{\sqrt{\alpha_t}} \nabla_{\boldsymbol{x}_t} \boldsymbol{m} \quad (23)$$

Putting all of them together we have the analytical formula of the conditional correction.

$$\nabla_{\boldsymbol{x}_t} \log p_{\boldsymbol{y}|t}(\boldsymbol{y}|\boldsymbol{x}_t) \approx \nabla_{\boldsymbol{x}_t} \boldsymbol{m}(\boldsymbol{x}_t) H^\top \left(HC(\boldsymbol{x}_t)H^\top + \sigma_y^2 I\right)^{-1} \left(\boldsymbol{y} - H\boldsymbol{m}(\boldsymbol{x}_t)\right) \quad (24)$$

Here, $\nabla_{\boldsymbol{x}_t}$ acts only on $\boldsymbol{m}$. This enables the conditional generation without extra training or data for guidance.

Crystal Structure Analysis in X-ray Diffraction Experiments

The X-ray diffraction intensity measured in the experiment changes with scattered direction. Scattering occurs only in directions satisfying Bragg's law:

$$\boldsymbol{s} = \frac{\boldsymbol{S} - \boldsymbol{S_0}}{\lambda} = h\boldsymbol{a}^* + k\boldsymbol{b}^* + l\boldsymbol{c}^* \quad (25)$$

where $\boldsymbol{S_0}$ ($\boldsymbol{S}$) is the unit vector of the incident (scattered) direction, λ is the X-ray wavelength, $(\boldsymbol{a}^*, \boldsymbol{b}^*, \boldsymbol{c}^*)$ are the reciprocal lattice vectors (i.e., bi-orthogonal system of the lattice vector $L = (\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})$). The three integers (h, k, l) specify the direction.

The amplitude of X-ray scattered toward $\boldsymbol{s}$ by an electron at position $\boldsymbol{r}$ is proportional to the real part of $e^{2\pi i\, \boldsymbol{r}\cdot\boldsymbol{s}}$. Hence X-ray amplitude from a crystal is proportional to the Fourier transform of the electron density of the crystal.

$$amp \sim \int \rho_{crys}(\boldsymbol{r})\, e^{2\pi i\, \boldsymbol{r}\cdot\boldsymbol{s}} d\boldsymbol{r} \quad (26)$$

When the ν-th atom with the electron density $\rho_\nu(\boldsymbol{r})$ is located at $\boldsymbol{r}_\nu$, the electron density in the unit cell is sum of the convolution

$$\rho_{cell}(\boldsymbol{r}) = \sum_{\nu=1}^{N} \rho_\nu(\boldsymbol{r}) * \delta(\boldsymbol{r} - \boldsymbol{r}_\nu) \quad (27)$$

and the electron density of the crystal is the convolution of it with the three comb functions.

$$\rho_{crys}(\boldsymbol{r}) = \rho_{cell}(\boldsymbol{r}) * \sum_{n_1} \delta(\boldsymbol{r} - n_1\boldsymbol{a}) * \sum_{n_2} \delta(\boldsymbol{r} - n_2\boldsymbol{b}) * \sum_{n_3} \delta(\boldsymbol{r} - n_3\boldsymbol{c}) \quad (28)$$

Because of the convolution theorem the amplitude is

$$amp \sim F(\boldsymbol{s}) \sum_{n_i} \delta(\boldsymbol{s} - n_1\boldsymbol{a}^*)\delta(\boldsymbol{s} - n_2\boldsymbol{b}^*)\delta(\boldsymbol{s} - n_3\boldsymbol{c}^*) \quad (29)$$

$$F(\boldsymbol{s}) = \sum_{\nu=1}^{N} f_\nu\,(s) e^{2\pi i\, x_\nu \cdot (h,k,l)} \quad (30)$$

which explains Bragg's law. Here, f_v is the element-specific constant called the atomic scattering factor and $\boldsymbol{x}_v$ is the fractional coordinate. In short, the X-ray scattering by a crystal is characterized with the so-called the structure factor (Woolfson 1997).

$$F(hkl) = \sum_{v=1}^{N} f_v e^{2\pi i\, \boldsymbol{x}_v \cdot (h,k,l)} \tag{31}$$

Since *F* is the Fourier transform of the unit cell electron density, determining complex *F* (including its phase) solves the atomic coordinates. However, only intensity, which is proportional to $|F(hkl)|^2$, is measurable in experiments. The absence of phase information is known as the "phase problem" and is a major obstacle in crystal structure analysis.

The direct method is a primary approach for solving the phase problem, effective for simple structures composed of light elements. Using highly-probable mathematical relationships among phases, the method first generates several reliable initial phase sets from the observed structure factor amplitudes. New phases are then estimated from these, and the phase sets are extended. During this iteration the reliability of each trial solution is quantified, and the best solution is selected.

The structural parameters of this initial model are then refined. Due to thermal vibration, the atoms are Gaussian distributed at the equilibrium position. This effect is measured as the temperature factor in experiment, and the isotropic/anisotropic displacement parameter is estimated. Since the scattering from hydrogen atom is weak, the position is estimated from positions of other atoms. The whole structure is then refined by minimizing the difference between the measured and the predicted intensity.

Proposed Method

Determining the structure from X-ray diffraction experiments requires both theoretical and empirical knowledge; an initial structure is created from diffraction intensities and chemical knowledge, and it is refined by minimizing the error in the predicted intensities from the measured ones. We would like to solve the issue by using a diffusion model that learns empirical priors about the crystal structure, and the X-ray diffraction intensity as a theoretical constraint. Since the structure determines the diffraction intensity via Eq. (31), the TMPD-like classifier-guidance is used. We derive the Tweedy formula under periodic condition. This enables us to predict what diffraction image the structure in the reverse process will eventually give. The conditional score indicates the direction of the correct structure in the reverse diffusion process.

Tweedy Formula under periodicity

The fractional coordinate domain $[0,1)^{3\times N}$ forms a quotient space $R^{3\times N}/Z^{3\times N}$ due to the periodicity of crystals. As in DiffCSP, the transition from x_0 to x_t is given by WN distribution:

$$q(x_t|x_0) = \frac{1}{\sqrt{2\pi\sigma_t^2}} \sum_{Z=-\infty}^{+\infty} \exp\left(-\frac{(x_t - x_0 + Z)^2}{2\sigma_t^2}\right) \tag{32}$$

In analogy with usual diffusion model the transition probability from x_t to x_0 is assumed to WN distribution:

$$p(x_0|x_t) = \frac{1}{\sqrt{2\pi c_t}} \sum_{Z=-\infty}^{+\infty} \exp\left(-\frac{(x_0 - m_t + Z)^2}{2c_t}\right) \tag{33}$$

Here, we call $m_t = m(x_t)$ and $c_t = c(x_t)$ as the mean and the variance, respectively. Now we derive Tweedy formula on Torus. As explained before, it relates the moment with the score and the derivative. However, the mean depends on the arbitrary-chosen origin and is ill-defined on Torus, and it is not evident that such formula exists. We show that the derivative of the normalization condition of p,

$$\frac{\partial^n}{\partial x_t^n} \int_{-\infty}^{\infty} p(x_0|x_t) dx_0 = 0 \tag{34}$$

yields the corresponding formulas.

Below we use physicist's notation of the derivative, $\partial^n f = \partial^n f / \partial x_t^n$, $q = q(x_t|x_0)$, $p = p(x_0|x_t)$, $p_t = p(x_t)$, and $p_0 = p(x_0)$. Using Bayes' theorem $p = qp_0/p_t$, after tedious calculation we have

$$\partial p = \partial\left(\frac{qp_0}{p_t}\right) = p(\partial \log q - \partial \log p_t) \tag{35}$$

The integral of this equation is zero, which results in the first-order formula.

$$\int_0^1 p(x_0|x_t)\, \partial \log q\,(x_t|x_0) dx_0 = \partial \log p\,(x_t) \tag{36}$$

Similarly, the equation of the second derivative

$$\begin{aligned} &\frac{\partial^2 p}{p} \\ &= \frac{1}{p}\partial^2\left(\frac{qp_0}{p_t}\right) \\ &= \partial^2 \log q + (\partial \log q)^2 - 2(\partial \log q)(\partial \log p_t) \\ &\qquad -\partial^2 \log p_t + (\partial \log p_t)^2 \end{aligned} \tag{37}$$

results in

$$\begin{aligned} &\int_0^1 p(\partial^2 \log q + (\partial \log q)^2) dx_0 \\ &\qquad = \partial^2 \log p_t + (\partial \log p_t)^2 \end{aligned} \tag{38}$$

The score and its derivative in the RHS are approximated by the NN model, and the derivative of q in the left-hand side (LHS) is known. Hence, these identities establish the relation between score and its derivative and $m(x_t)$ and $c(x_t)$

in p. In other words, given x_t in the reverse diffusion, we know the final distribution via Eq. (33). Since this is an equation that cannot be solved analytically, we devise a method to solve it numerically. Details is presented in Appendix A.

Conditional Score

Eq. (33) indicates that the distribution of the v-th atom in a unit is specified by the Normal, $\mathcal{N}(\boldsymbol{x}_v; \boldsymbol{m}_v, \mathrm{diag}\, \boldsymbol{c}_v)$. Here, $\boldsymbol{m}_v \in R^3$ is the mean of the three fractional coordinates and $\boldsymbol{c}_v$ the variance. Replacing $\delta(\boldsymbol{r} - \boldsymbol{r}_v)$ with it leads to the structure factor.

$$F_{calc}(hkl) = \sum_{v=1}^{N} f_v e^{2\pi i\, \boldsymbol{m}_v \cdot (h,k,l) - 2\pi^2 \boldsymbol{c}_v \cdot (h^2,k^2,l^2)} \quad (39)$$

We dropped the unit cell volume of no interest. The square gives the predictive intensity,

$$I_{calc}(hkl) = \sum_{v,\xi=1}^{N} f_v f_\xi e^{2\pi i\, (\boldsymbol{m}_v - \boldsymbol{m}_\xi)\cdot(h,k,l) - 2\pi^2 (\boldsymbol{c}_v + \boldsymbol{c}_\xi)\cdot(h^2,k^2,l^2)} \quad (40)$$

Let I_{exptl} be the measured diffraction intensity, which is ideally,

$$I_{exptl}(hkl) = \sum_{v,\xi=1}^{N} f_v f_\xi e^{2\pi i\, (\boldsymbol{r}_v - \boldsymbol{r}_\xi)\cdot(h,k,l)} \quad (41)$$

and I_{calc} the predicted diffraction intensity from x_t. We assume that the difference, which corresponds to "measurement error", is the Gaussian noise:

$$I_{calc}(hkl) - I_{exptl}(hkl) = u, u \sim \mathcal{N}(0, \sigma_y^2 I) \quad (42)$$

Assume that measurement is done only for $N_h \times N_k \times N_l$ intensities, which constitutes of our condition y. The probability that y is realized given x_t is calculated as follows:

$$p(y|x_t) = \prod_{h,k,l} \exp\left(-\frac{\left(I_{calc}(hkl) - I_{exptl}(hkl)\right)^2}{2\sigma_y^2} \right) \quad (43)$$

Thus, the conditional score is calculated as follows:

$$\frac{\partial}{\partial x_t} \log p\,(y|x_t) = -\frac{1}{\sigma_y^2} \sum_{h,k,l} \left(I_{calc} - I_{exptl}\right) \frac{\partial I_{calc}}{\partial x_t} \quad (44)$$

We denote as $\partial \log p\,(y|X_t)/\partial X_t \in R^{3\times N}$ collection of them at all coordinates. The reverse diffusion process is then represented by the following equation:

$$X_{t-1} = w(X_t + \zeta_t \tilde{\boldsymbol{\epsilon}}_{\boldsymbol{X}} + \bar{\zeta}_t \boldsymbol{\epsilon}_{\boldsymbol{X}}) \quad (45)$$

$$\tilde{\boldsymbol{\epsilon}}_{\boldsymbol{X}} = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{X}}(\mathcal{M}_t, t) + \frac{\partial}{\partial X_t} \log p\,(y|X_t) \quad (46)$$

Here, $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{X}}(\mathcal{M}_t, t)$ is an approximated score without condition, $\zeta_t = \sigma_t^2 - \sigma_{t-1}^2$, and $\bar{\zeta}_t = (\sigma_{t-1}/\sigma_t)\sqrt{\zeta_t}$. For the crystal lattice vector L, the same reverse diffusion step as in DiffCSP is applied. Eq. (44) contains various derivatives, and their numerical calculation method is summarized in Appendix A.

Experiment

This section presents three experiments to explore the effectiveness of the proposed method, along with their results.

Definitions and Experimental Parameters

The matching rate measures how well the generated crystal structures align with the true crystal structures. As previous work we evaluated it using the StructureMatcher class in pymatgen (Ong et al. 2013), which determines matches based on three thresholds:

- angle_tol: The maximum angular difference (in degrees) allowed between corresponding lattice angles for two structures to be considered equivalent.
- ltol: The tolerance for lattice vector lengths. It allows slight variations in lattice dimensions.
- stol: The site tolerance, which determines how far atomic positions can deviate for two structures to match.

These thresholds define the level of similarity required for a match. The matching rate is defined as the proportion of sampled crystal structures that match the true structures.

In all experiments, Miller indices (h, k, l) was taken from (-2, -1, 0, 1, 2). As a result, there are a total of 5^3 diffraction intensities to reproduce.

The σ_y parameter in Eq. (44) is dynamically determined during sampling to ensures that the maximum absolute values of $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{X}}(\mathcal{M}_t, t)$ and the conditional score $\partial \log p\,(y|X_t)/\partial X_t$ keep a specific ratio. We call this hyperparameter the score ratio, which differs in each experiment. Score ratio = 1:3 means that the maximum absolute values of $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{X}}(\mathcal{M}_t, t)$ and the conditional score are 1:3. We found empirically that this adjustment stabilizes and improves the reverse diffusion process. Other training details are provided in the Appendix.

Selective Generation of Crystals from Diffraction Intensity

To verify the effectiveness of the proposed method, we conducted a simple experiment: from two crystal structures with the same chemical composition but different atomic arrangements, generate the one specified with the X-ray diffraction intensities.

Dataset As shown in Figure 1, two artificial crystal structures, Crystal-1 and Crystal-2, were prepared. Both are the

Table 1: Comparison of generation accuracy without/with the conditional score.

	DiffCSP	**Ours**
Crystal-1	15.7%	**95.6%**
Crystal-2	7.5%	**95.5%**

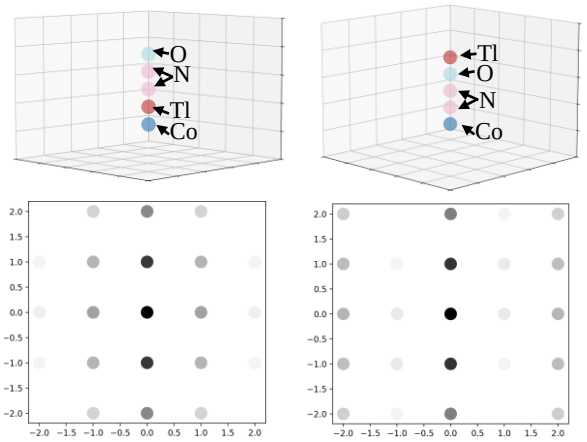

Figure 1: Crystal-1 (top left) has the sequence [O, N, N, Tl, Co], while Crystal-2 (top right) has the sequence [Tl, O, N, N, Co]. The bottom shows the corresponding X-ray diffraction intensities (h=0).

same cubic lattice with constant 4.24596 Å, have the same chemical composition CoN_2OTl, but the atomic sequences differ: Crystal-1 has the sequence [O, N, N, Tl, Co], while Crystal-2 has the sequence [Tl, O, N, N, Co]. Gaussian noise $\mathcal{N}(0, 0.01^2)$ was added to each atomic position to augment the dataset, creating a total of 10,000 samples (5,000 for each type).

Sampling Sampling was performed using both DiffCSP, which does not use conditioning, and the proposed method. The score ratio was set to 1:4. The X-ray diffraction intensity of Crystal-1/2 was calculated using the atomic scattering factors in CrysFML (Rodriguez-Carvajal and Gonzalez-Platas 2006). The thresholds for matching rate were set with stol=0.05, angle_tol=10, and ltol=0.3.

The results are summarized in Table 1. Without conditions, ideally either Crystal-1 or 2 should be generated and the match ratio would be 50%, but it was much lower in our matching threshold. By adding diffraction constraint it drastically improved to 96%. We can say that our method not only biased the generation toward one of the two structures but also made fine adjustments based on subtle differences in X-ray diffraction intensity, enabling it to handle random noise effectively.

Generation of Unlearned Crystals

In the first experiment, we demonstrated that the proposed method can differentiate between learned structures. However, in X-ray diffraction experiments, it is often necessary to determine unknown structures that have not been encountered during training. This experiment aims to verify whether the method can generate structures which are not present in the training data. To this end, a new structure, Crystal-3, with the same lattice parameter and the chemical

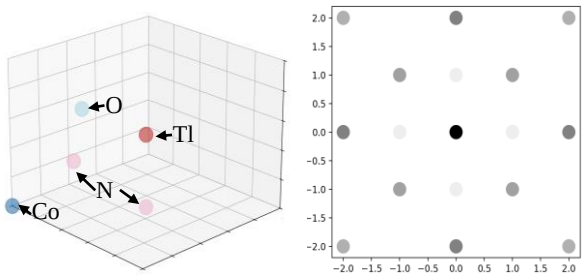

Figure 2: Crystal-3 (left) has the same lattice parameter and the composition as Crystal-1/2, but a different structure. The right shows the X-ray diffraction intensity (h=0).

composition as Crystal-1/2 but different atomic arrangements and X-ray diffraction intensities, was introduced. Crystal-3 was taken from in the Perov-5 dataset (Castelli et al. 2012a; Castelli et al. 2012b). Crystal-1 and Crystal-3 were used as training data, and the ability to generate Crystal-2 was tested by conditioning on its X-ray diffraction intensities. Gaussian noise $\mathcal{N}(0, 0.01^2)$ was added to each coordinate to augment the dataset, creating a total of 10,000 samples (5,000 for each type).

Sampling We generate 10,000 variants of the Crystal-2 structures by adding random noise $\mathcal{N}(0, 0.01^2)$, and calculate the X-ray diffraction intensities. Structures are generated with and without these intensity constraints. The score ratio was set to 1:4. The thresholds for matching rate were set with angle_tol=10 and ltol=0.3, while three values (0.05, 0.1, 0.5) were used for stol.

Results Sampled structures are shown in Figure 3, and the matching rate for each stol values are summarized in Table 2. The matching rate with stol=0.05 was 0% for both methods. However, when the stol threshold was increased, our method achieved a higher matching rate. However, no crystals with the same arrangement as Crystal-2 [Tl, O, N, N, Co] were observed.

Although crystals with the same atomic arrangement as Crystal-2 were not obtained, the X-ray diffraction intensities of the sampled crystals were similar to those of the true structure. X-ray diffraction intensities are highly dependent on the positions of atoms with large scattering factors, i.e., heavier atoms. Co and Tl atoms have much more electrons than N and O. Hence, during reverse diffusion, Co and Tl

Table 2: Comparison of generation accuracy between DiffCSP and ours, showing the generation accuracy for both Crystal-1 and Crystal-2. The stol parameter is the threshold for the StructureMatcher class, and the matching accuracy was measured at three different threshold levels.

stol	DiffCSP	**Ours**
0.05	0.00%	0.00%
0.1	0.00%	**0.62%**
0.5	8.03%	**96.21%**

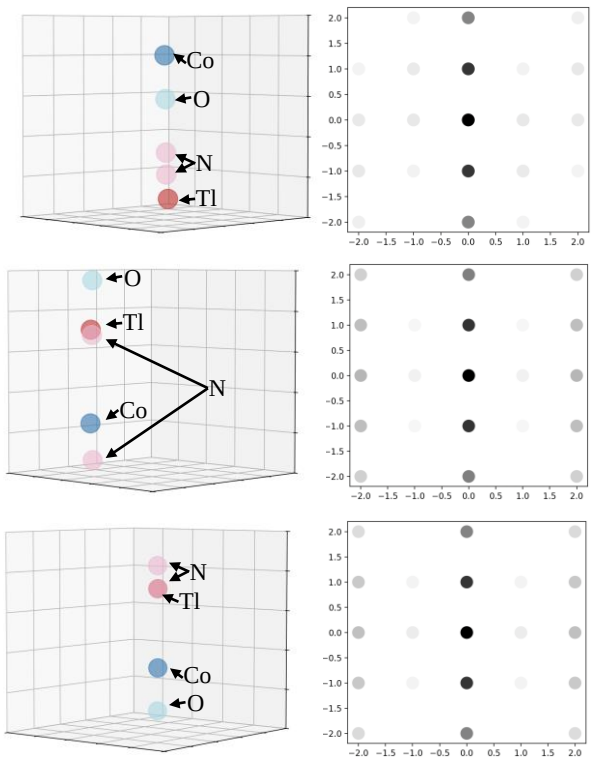

Figure 3: The left column shows the sampled crystal structures, and the right show the corresponding X-ray diffraction intensities (h=0).

atoms are more mobile than N and O. This explains the appearance of structures such as [Tl, N, N, O, Co], where the Co atom shifted closer to the O atom than to the Tl atom, or the sample where the Tl atom moved away from the Co atom, overlapped with the N atom.

Datasets Containing Various Elements

Following Jiao et al. (2023), we also evaluated generation accuracy on well-known crystal datasets: Perov-5 (Castelli et al. 2012a; Castelli et al. 2012b), Carbon-24 (Pickard 2020), MP-20 and MPTS-52 (Jain et al. 2013). For each dataset, the data were divided into training, validation, and test sets. During training, both the training and validation sets were used, while the test set was used to measure the matching rate.

Dataset Perov-5 includes 18,920 perovskite materials with similar structures, each containing 5 atoms per unit cell. Carbon-24 contains 10,153 carbon materials with 6-24 atoms in each unit cell. MP-20, derived from the Materials Project (Jain et al. 2013), consists of 45,231 stable inorganic materials, mostly experimentally synthesized, with up to 20 atoms per unit cell. MPTS-52 is an extension of MP-20, containing 40,476 structures with up to 52 atoms, sorted by the earliest publication year. Perov-5, Carbon-24, and MP-20 were split following Xie et al. (2021) at a 60-20-20 ratio.

Table 3: Comparison of generation accuracy between DiffCSP and our method across the four datasets.

Method	Perov-5	Carbon-24	MP-20	MPTS-52
DiffCSP	48.24%	8.47%	52,21%	9.62%
Ours	**61.56%**	**50.05%**	**52.58%**	**10.76%**

Following Jiao et al. (2023), MPTS-52 was divided into 27,380/5,000/8,096 samples for training, validation, and testing based on chronology.

Sampling was performed using both DiffCSP, which does not use conditioning, and the proposed method. The score ratio was set to 1:3. Following Jiao et al. (2023), thresholds for matching rate were set with stol=0.5, angle_tol=10, and ltol=0.3.

Results The matching rates for each dataset are presented in Table 3. As shown, matching rates improved moderately in Perov-5, significantly in Carbon-24, and slightly in MP-20 and MPTS-52. The highest matching rate for Carbon-24 is due to the fact that this is the only dataset composed of a single element. In the second experiment, it was observed that lighter atoms with smaller scattering factors tend to stagnate in motion during generation. In these datasets all atoms seem to move to the correct position in the same way, result in better structures.

From the detailed analysis in the Appendix, it was observed that datasets with fewer atoms in the unit cell exhibit higher matching rates, and that matching rates tend to decrease as the number of atoms increases. As this was observed both with and without conditions, it is considered a characteristic of the diffusion model. The number of randomly generated initial structures can be predicted from the entropy of the ideal gas mixture. The number of initial structures increases exponentially with the number of atoms. Moreover, as the number of atoms increases, the diffraction intensity becomes less sensitive to the movement of the atoms, making position adjustment more difficult.

Conclusion

The diffusion model, which has achieved great success in image generation, is now being applied to the field of science, such as structure generation of materials and crystals. The powerful ability of the diffusion model to learn empirical prior is expected to be applied to scientific inference. In the field of conventional X-ray crystallography, the empirical prior and the physical theory of diffraction are used in combination to determine the structure from experimentally observed diffraction intensities. As an example of the adaptation of the diffusion model to such scientific reasoning, this paper proposes a method for generating crystal structures given X-ray diffraction intensities. Such conditional

generation has been accomplished in image domain by learning a large amount of image and text prompt pairs. However, the measured structure is not included in the training data; only the physical theory that determines the diffraction intensity is known.

Using the torus-based diffusion process and the newly-derived Tweedy formula on it, we derived analytically the conditional score, or the guidance, for the crystal generation without label learning by leveraging a pre-trained diffusion model. We applied this method to several systems and showed that structures that reproduce diffraction intensities are preferentially generated. At the same time, it was found to be difficult when the unit cell contains many atoms or when heavy and light atoms are mixed. The former may be related to the lack of sufficient empirical prior. Since the crystal structure is a stable structure, the unconditional score is expected to be close to the force field of the material. Some of the generated samples had chemically invalid structures, such as very close atomic contact, which a force fields never generates.

The latter may be due to the fact that the light atom has only a small diffraction intensity and enough guidance does not work. We need to design an appropriate loss function or guidance scale. In fact, the guidance scale (score ratio) significantly affecting generation accuracy, as demonstrated in Appendix D. These advancements will help refine the proposed method and broaden its applicability to more complex structures, improving its effectiveness in X-ray diffraction-based structure determination.

References

Bortoli, V. D.; Mathieu, E.; Hutchinson, M. J.; Thornton, J.; Teh, Y. W.; and Doucet, A. 2022. Riemannian score-based generative modelling. In Proceedings of the 35th International Conference on Neural Information Processing Systems 2406-2422. Red Hook, NY: Curran Associates Inc.

Boys, B.; Girolami, M.; Pidstrigach, J.; Reich, S.; Mosca, A.; and Akyildiz, O. D. 2024. Tweedie Moment Projected Diffusions for Inverse Problems. *Transactions on Machine Learning Research*: 2835-8856. New York, NY: Journal of Machine Learning Research Inc.

Castelli, I. E.; Olsen, T.; Datta, S.; Landis, D. D.; Dahl, S.; Thygesen, K. S.; and Jacobsen, K. W. 2012. Computational screening of perovskite metal oxides for optimal solar light capture. *Energy & Environmental Science*, 5(2):5814–5819. London, ENG: Royal Society of Chemistry.

Castelli, I. E.; Landis, D. D.; Thygesen, K. S.; Dahl, S.; Chorkendorff, I.; Jaramillo, T. F.; and Jacobsen, K. W. 2012. New cubic perovskites for one-and two-photon water splitting using the computational materials repository. *Energy & Environmental Science*, 5(10): 9034–9043. London, ENG: Royal Society of Chemistry.

Dhariwal, P.; and Nichol, A. 2021. Diffusion Models Beat GANs on Image Synthesis. In Proceedings of the 34th International Conference on Neural Information Processing Systems 8780-8794. Red Hook, NY: Curran Associates Inc.

Gasteiger, J.; Becker, F.; and Günnemann, S. 2021. GemNet: Universal Directional Graph Neural Networks for Molecules. In Proceedings of the 34th International Conference on Neural Information Processing Systems 6790-6802. Red Hook, NY: Curran Associates Inc.

Gasteiger, J.; Giri, S.; Margraf, J. T.; and Günnemann, S. 2020. Fast and Uncertainty-Aware Directional Message Passing for Non-Equilibrium Molecules. arXiv preprint. arXiv:2011.14115[cs.LG]. Ithaca, NY: Cornell University Library.

Gasteiger, J.; Groß, J.; and Günnemann, S. 2020. Directional Message Passing for Molecular Graphs. arXiv preprint. arXiv:2003.03123[cs.LG]. Ithaca, NY: Cornell University Library.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In Proceedings of the 33rd International Conference on Neural Information Processing Systems 6840-6851. Red Hook, NY: Curran Associates Inc.

Ho, J., and Salimans, T. 2022. Classifier-Free Diffusion Guidance. arXiv preprint. arXiv:2207.12598[cs.LG]. Ithaca, NY: Cornell University Library.

Jain, A.; Ong, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; and Persson, K. A. 2013. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials*, 1(1):011002. Melville, NY: American Institute of Physics.

Jiao, R.; Huang, W.; Lin, P; Han, J.; Chen, P.; Lu, Y.; and Liu Y. 2023. Crystal Structure Prediction by Joint Equivariant Diffusion. In Proceedings of the 36th International Conference on Neural Information Processing Systems 17464-17497. Red Hook, NY: Curran Associates Inc.

Jing, B.; Corso, G.; Chang, J.; Barzilay, R.; and Jaakkola, T. 2022. Torsional diffusion for molecular conformer generation. arXiv preprint. arXiv:2206.01729[cs.LG]. Ithaca, NY: Cornell University Library.

Juan Rodriguez-Carvajal, J.; Gonzalez-Platas, J. 2006. CrysFML, http://forge.ill.eu/projects/show/crysfml.

Ong, S. P.; Richards, W. D.; Jain, A.; Hautier, G.; Kocher, M.; Cholia, S.; Gunter, D.; Chevrier, V. L.; Persson, K. A.; and Ceder, G. 2013. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319. Amsterdam, Noord-Holland: Elsevier B.V.

Pan, H.; Ganose, A. M.; Horton, M.; Aykol, M.; Persson, K. A.; Zimmermann, N. E. R.; and Jain, A. 2021. Benchmarking Coordination Number Prediction Algorithms on Inorganic Crystal Structures. *Inorganic Chemistry*, 60(3): 1590–1603. Washington, DC: American Chemical Society.doi:10.1021/acs.inorgchem.0c02996

Pickard, C. J. 2020. AIRSS Data for Carbon at 10GPa and the C+N+H+O System at 1GPa. Materials Cloud Archive. https://archive.materialscloud.org/record/2020.0026/v1. Accessed: 2024-10-06.

Song, Y., and Ermon, S. 2020. Improved techniques for training score-based generative models. In Proceedings of the 33rd on Neural Information Processing Systems 12438-12448. Red Hook, NY: Curran Associates Inc.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole B. 2021. Score-Based Generative Modeling through

Stochastic Differential Equations. arXiv preprint. arXiv:2011.13456[cs.LG]. Ithaca, NY: Cornell University Library.

Xie, T.; Fu, X.; Ganea, O.; Barzilay, R.; and Jaakkola, T. 2021. Crystal Diffusion Variational Autoencoder for Periodic Material Generation. arXiv preprint. arXiv:2110.06197[cs.LG]. Ithaca, NY: Cornell University Library.

Woolfson, M. M., 1997. An introduction to X-ray crystallography, second edition, Cambridge University Press.

Appendix

A. Solving Periodic Tweedy Formula and Calculating Conditional Score

During the reverse diffusion process we have to solve Eq. (36) and Eq. (38) to determine $m(x_t)$ and $c(x_t)$. The analytical solution is not known and we used the following numerical procedure. First, since the transition probability $q(x_t|x_0)$ only depends on $x_t - x_0$, we write $\log q(x_t|x_0) = g(x_t - x_0)$. This is a periodic function

$$g(y,t) = \log\left(\frac{1}{\sqrt{2\pi\sigma_t^2}}\sum_{Z=-\infty}^{+\infty}\exp\left(-\frac{(y+Z)^2}{2\sigma_t^2}\right)\right) \tag{A1}$$

and we examine it in the domain $k - \frac{1}{2} \le y \le k + 1/2$ for $k \in Z$.When σ_t is large, this g is a smooth function which is suitable for numerical analysis. However as $\sigma_t \to 0^+$ it may become singular and require separate analysis. In fact, at this limit only $Z = -k$ term remains and $dg/dy \approx -(y-k)/\sigma^2$, meaning g becomes a sawtooth wave. Below we denote $g' = dg(x_t - x_0)/dx_t$. The LHS of Eq. (36) is

$$(L36) = \int_0^1 g'\frac{1}{\sqrt{2\pi c_t}}\sum_{Z=-\infty}^{+\infty}\exp\left(-\frac{(x_0 - m_t + Z)^2}{2c_t}\right)dx_0 = \frac{1}{\sqrt{2\pi c_t}}\int_{-\infty}^{\infty} g'(x_0)\exp\left(-\frac{(x_t - x_0 - m_t)^2}{2c_t}\right)dx_0 \tag{A2}$$

In the generation process, σ_t gradually approaches 0^+ and the sample becomes progressively clearer and sharper. We speculate that conditioning is most important at this later stage. Replacing g' with its limit, we have

$$(L36) = \sum_{k=-\infty}^{+\infty}\frac{-1}{\sigma^2\sqrt{2\pi c_t}}\int_{k-\frac{1}{2}}^{k+\frac{1}{2}}(x-k)\exp\left(-\frac{(x_t - x - m_t)^2}{2c_t}\right)dx = \sum_{k=-\infty}^{+\infty} J_k^{(1)} \tag{A3}$$

$$\sigma^2 J_k^{(1)} = \sqrt{\frac{c_t}{2\pi}}\left(e^{-K_+^2} - e^{-K_-^2}\right) + \frac{1}{2}\sqrt{\frac{c_t}{2}}(K_+ + K_-)(\mathrm{erf}(K_+) - \mathrm{erf}(K_-)) \tag{A4}$$

Here, the symbol is defined as $K_\pm = (k + m_t - x_t \pm 1/2)/\sqrt{2c_t}$. Similarly, using the asymptotic formula

$$\partial^2 \log q + (\partial \log q)^2 \approx \frac{1}{\sigma^2}\sum_l \delta\left(y - l + \frac{1}{2}\right) - \frac{1}{\sigma^2} + \frac{(y-k)^2}{\sigma^4} \tag{A5}$$

we have

$$(L38) = \sum_{k=-\infty}^{+\infty}\frac{1}{\sigma^4\sqrt{2\pi c_t}}\int_{k-\frac{1}{2}}^{k+\frac{1}{2}}(x-k)^2\exp\left(-\frac{(x_t - x - m_t)^2}{2c_t}\right)dx = \sum_{k=-\infty}^{+\infty} J_k^{(2)} \tag{A6}$$

$$\sigma^4 J_k^{(2)} = \sqrt{\frac{c_t}{2\pi}}\left(-\frac{1}{2}\left(e^{-K_+^2} + e^{-K_-^2}\right) + \sqrt{\frac{c_t}{2}}(K_+ + K_-)\left(e^{-K_+^2} - e^{-K_-^2}\right)\right) + \frac{c_t}{4}(2 + (K_+ + K_-)^2)(\mathrm{erf}(K_+) - \mathrm{erf}(K_-)) \tag{A7}$$

We truncated the sums of Eqs. (A3) and (A6) in the range $-10 \le k \le 10$. The RHS of Eq. (38) contains the derivative of the score, which is very difficult to approximate, and we simply ignore it as in previous study. Given RHS of Eqs. (36) and (38) we find m_t and c_t that satisfy the equation by the table search.

As shown in Eq. (44) the conditional score involves $\partial I_{calc} = (\partial I_{calc}/\partial m_t)\partial m_t + (\partial I_{calc}/\partial c_t)\partial c_t$. We first derive formulas of $\partial I_{calc}/\partial m_t$ and $\partial I_{calc}/\partial c_t$.

$$\frac{I_{calc}}{\partial \boldsymbol{m_v}} = -4\pi \boldsymbol{h} f_v \sum_{\xi} f_\xi \sin\left(2\pi\left(\boldsymbol{m_v} - \boldsymbol{m_\xi}\right)\cdot \boldsymbol{h}\right) e^{-2\pi^2(\boldsymbol{c_v} + \boldsymbol{c_\xi})\cdot(h^2,k^2,l^2)} \tag{A8}$$

$$\frac{I_{calc}}{\partial \boldsymbol{c_v}} = -4\pi^2(h^2,k^2,l^2) f_{\boldsymbol{v}} \sum_{\xi} f_\xi \cos\left(2\pi\left(\boldsymbol{m_v} - \boldsymbol{m_\xi}\right)\cdot \boldsymbol{h}\right)\ e^{-2\pi^2(\boldsymbol{c_v} + \boldsymbol{c_\xi})\cdot(h^2,k^2,l^2)} \tag{A9}$$

Here, $\boldsymbol{m_v} = \left(m_{vx}, m_{vy}, m_{vz}\right) \in R^3$ represents the mean of each of the three-dimensional coordinates for the *v*-th atom at time t, $\boldsymbol{c_v} = \left(c_{vx}, c_{vy}, c_{vz}\right) \in R^3$ the corresponding variance, and $\boldsymbol{h} = (h,k,l) \in R^3$.

To calculate ∂m_t and ∂c_t, we take the variation of the Tweedy formulas with respect to x_t. With the variation δx_t RHS of Eq. (36) changes by $\left(\partial^2 \log p\,(x_t)\right)\delta x_t$, and RHS of Eq. (38) changes by $\left(\partial^3 \log p\,(x_t) + 2\,\partial^2 \log p\,(x_t)\,\partial \log p\,(x_t)\right)\delta x_t$. On LHS, $\delta c = \partial c \delta x_t$ and $\delta m = \partial m \delta x_t$. Then,

$$\delta(L36) = \left(\frac{\partial(L36)}{\partial c}\partial c + \frac{\partial(L36)}{\partial m}\partial m + \frac{\partial(L36)}{\partial x_t}\right)\delta x_t \tag{A10}$$

$$\delta(L38) = \left(\frac{\partial(L38)}{\partial c}\partial c + \frac{\partial(L38)}{\partial m}\partial m + \frac{\partial(L38)}{\partial x_t}\right)\delta x_t \tag{A11}$$

Thus, ∂m_t and ∂c_t are the solution of the following system of equations.

$$\begin{pmatrix} \frac{\partial(L36)}{\partial m} & \frac{\partial(L36)}{\partial c} \\ \frac{\partial(L38)}{\partial m} & \frac{\partial(L38)}{\partial c} \end{pmatrix} \begin{pmatrix} \partial m \\ \partial c \end{pmatrix} = \begin{pmatrix} \partial^2 \log p_t + \frac{\partial(L36)}{\partial m} \\ \partial^3 \log p_t + 2(\partial^2 \log p_t)(\partial \log p_t) + \frac{\partial(L38)}{\partial m} \end{pmatrix} \tag{A12}$$

Here we used $\partial(L36)/\,\partial x_t = -\,\partial(L36)/\,\partial m$, derived from (A2). Other derivatives in these equations are given as,

$$\sigma^2 \frac{\partial(L36)}{\partial m} = \frac{-1}{2\sqrt{2\pi c}} \sum_{k=-\infty}^{+\infty} \left(e^{-K_+^2} + e^{-K_-^2}\right) + \frac{1}{2} \tag{A13}$$

$$\sigma^2 \frac{\partial(L36)}{\partial c} = \frac{1}{8c\sqrt{2\pi c}} \sum_{k=-\infty}^{+\infty} \left((1+4c)\left(e^{-K_+^2} - e^{-K_-^2}\right) + \frac{1}{2}\sqrt{\frac{c_t}{2}}\,(K_+ + K_-)\left(e^{-K_-^2} + e^{-K_+^2}\right) \right) \tag{A14}$$

$$\sigma^4 \frac{\partial(L38)}{\partial m} = \frac{1}{2c\sqrt{2\pi c}} \sum_{k=-\infty}^{+\infty} \frac{8c+1}{4} \frac{1}{\sqrt{2\pi c}} \left(e^{-K_-^2} - e^{-K_+^2}\right) + 2k(\mathrm{erf}(K_+) - \mathrm{erf}(K_-)) + m - x_t \tag{A15}$$

$$\sigma^4 \frac{\partial(L38)}{\partial c} = \frac{-1}{4c\sqrt{2\pi c}} \sum_{k=-\infty}^{+\infty} \frac{8c+1}{4} \left(e^{-K_+^2} + e^{-K_-^2}\right) + \frac{1}{2}\sqrt{\frac{c_t}{2}}\,(K_+ + K_-)\left(e^{-K_+^2} - e^{-K_-^2}\right) + \frac{1}{2} \tag{A16}$$

B. Hyper-parameters and Training Details

In all experiments, the model was configured with 6 layers and 512 hidden states. The dimension of the Fourier embedding was set to k = 256. We applied a cosine scheduler with $s = 0.008$ to control the variance of the DDPM process on L_t, and an exponential scheduler with $\sigma_1 = 0.005$ and $\sigma_T = 0.5$ to control the noise scale of the score matching process on X_t. The diffusion step was set to T = 1000. In the first and the second experiments and experiment [D], training was conducted for 40,000 epochs. In the third experiment, the number of epochs varied depending on the dataset: 3500, 4000, 1000, and 1000 epochs for Perov-5, Carbon-24, MP-20, and MPTS-52, respectively. The optimizer used was Adam with an initial learning rate of 1×10^{-3}, along with a Plateau scheduler with a decay factor of 0.6 and a patience of 30 epochs. In the first three experiments, both the lattice vectors *L* and the atomic coordinates *X* were considered as targets for generation, and the score network was trained to optimize both scores. In Experiment [D], however, to exclude the influence of lattice vectors *L*, they were not considered as targets for generation. Instead, the training focused solely on the atomic coordinates *X*, and as a result, the score network did not learn the score for lattice vectors. The inputs to the score network, $\mathcal{M}_t$, were defined as $[L_0, X_t, A]$, where L_0 represents the initial lattice vectors. For sampling, the step size γ was set to 1×10^{-5} for all cases. For reproducibility, the variance in the reverse diffusion process, which introduces randomness, was set to zero for both lattice vectors *L* and atomic coordinates *X* for all cases.

C. Additional Results of the 3rd Experiment

This section describes the details of the third experiment for each dataset. The matching rate was analyzed for each composition formula in the Perov-5 dataset, while in other datasets, it was analyzed based on the number of atoms.

C-1. Details of Perov-5 sampling

Using composition formula we divide the test data into six groups: ABN_3, ABO_3, ABN_2O, $ABNO_2$, ABNOF, and ABO_2F. Here, A and B can be any metal atoms. We measured the matching rate for each group, and the results are shown in Table C1. Accuracy improves across all groups.

Table C1: matching rate of each formula group in Perov-5. #-sample represents the number of samples.

Group	#-sample	DiffCSP	**Ours**
ABN3	550	48.00%	**63.64%**
ABO3	553	50.27%	**65.82%**
ABN2O	530	46.42%	**58.87%**
ABNO2	539	44.90%	**63.64%**
ABNOF	574	51.39%	**58.36%**
ABO2F	515	49.13%	**59.81%**
ABSO2	524	47.33%	**60.69%**

C-2. Details of Carbon-24 sampling

With our learned weight and the matching tolerance the unconditional generation did not work, but condition by diffraction intensity saves this failure: a significant improvement in generation accuracy is observed for crystals with 6 atoms. As the number of atoms in the unit cell increases, the matching rate decreases. No matching crystal structures were obtained for 14 or more without condition, and 20 or more with the condition.

Table C2: matching rate dependence on the number of atoms (#-atoms) in Carbon-24 dataset.

#-atoms	#-sample	DiffCSP	**Ours**
6	705	14.75%	**73.62%**
8	526	10.27%	**57.41%**
10	317	2.84%	**38.49%**
12	200	1.50%	**16.50%**
14	110	0.00%	**22.73%**
16	84	0.00%	**11.90%**
18	37	0.00%	**13.51%**
20	24	0.00%	0.00%
22	17	0.00%	0.00%
24	10	0.00%	0.00%

C-3. Details of MP-20 sampling

Unconditional generation nicely worked on this dataset, especially samples with small number of atoms, as shown in Table C3. Conditional generation sometimes improves but sometimes not. As the number of atoms increases and the problem becomes more difficult, conditional generation begins to show superior results.

Table C3: matching rate of MP-20 dataset.

#-atoms	#-sample	DiffCSP	**Ours**
1	30	66.67%	66.67%
2	198	80.30%	**86.36%**
3	181	77.35%	**91.71%**
4	1,443	81.64%	**93.21%**
5	390	**85.90%**	81.54%
6	771	61.22%	**61.48%**
7	192	**54.69%**	51.04%
8	732	**53.28%**	46.45%
9	305	**58.69%**	37.05%
10	926	**63.61%**	52.48%
11	116	**32.76%**	23.28%
12	846	**37.00%**	36.29%
13	193	31.61%	**33.68%**
14	573	31.24%	**34.21%**
15	135	15.56%	**25.93%**
16	594	19.36%	**26.60%**
17	82	31.71%	**39.02%**
18	474	20.04%	**26.58%**
19	78	**26.92%**	25.64%

20	787	**36.47%**	0.00%
22	17	0.00%	0.00%
24	10	0.00%	0.00%

C-4. Details of MPTS-52 sampling

MPTS-52 is an extension of MP-20 dataset, and similar trends is observed in Table C4. For crystals with 33 or more atoms, no matching samples were obtained using DiffCSP, but our method successfully produced some matching samples.

Table C4: matching rate of MPTS-52 dataset.

#-atoms	#-sample	DiffCSP	**Ours**
1	6	66.67%	66.67%
2	6	33.33%	33.33%
3	26	42.31%	**65.38%**
4	80	46.25%	**53.75%**
5	136	56.62%	**58.09%**
6	216	**41.20%**	37.04%
7	64	**28.12%**	14.06%
8	289	21.80%	21.80%
9	168	**31.55%**	23.21%
10	434	**41.01%**	22.58%
11	104	9.62%	**13.46%**
12	414	18.60%	**26.09%**
13	66	6.06%	**18.18%**
14	212	7.55%	**8.02%**
15	51	3.92%	**7.84%**
16	260	2.69%	**10.77%**
17	74	17.57%	**20.27%**
18	281	7.83%	**8.90%**
19	35	5.71%	**8.57%**
20	596	3.86%	**14.60%**
21	80	1.25%	1.25%
22	319	0.63%	**3.45%**
23	52	0.00%	0.00%
24	577	9.53%	**12.82%**
25	30	0.00%	0.00%
26	167	0.00%	0.60%
27	37	2.70%	2.70%
28	463	2.38%	**3.89%**
29	23	0.00%	0.00%
30	217	0.00%	0.00%
31	19	0.00%	0.00%
32	321	0.31%	**1.56%**
33	10	0.00%	0.00%
34	139	0.00%	0.00%
35	16	0.00%	**6.25%**
36	380	0.00%	**1.05%**
37	16	0.00%	0.00%
38	134	0.00%	**1.49%**
39	16	0.00%	0.00%
40	323	0.00%	**0.93%**
41	10	0.00%	0.00%
42	152	0.00%	**0.66%**
43	7	0.00%	0.00%
44	266	0.00%	**0.38%**
45	19	0.00%	0.00%
46	109	0.00%	0.00%
47	5	0.00%	0.00%
48	294	0.00%	**0.34%**
49	10	0.00%	0.00%
50	96	0.00%	0.00%
51	2	0.00%	0.00%
52	269	0.00%	0.00%

D. Generation of Superlattice Structures

In Experiment 1, one of two toy structures was generated based on diffraction intensity. This experiment extends the scope to the realistic material, the Cu3Au alloy, which exhibits the low-temperature phase of the superlattice structure, and the solid solution alloys. The former is a periodic arrangement of atoms in which certain positions are preferentially occupied by minor atomic species. In this case one of four Cu atoms in the base FCC lattice is replaced with Au to make the unit cell, and this cell is periodically repeated. The experiment aims to generate the superlattice structure conditioned with the diffraction intensities. Additionally, we investigate how the score ratio affects generation accuracy.

Dataset Crystal structure was created as a supercell, connecting two FCC unit cells of Cu3Au (Figure D1). Hence, each sample contains 6 Cu atoms and 2 Au atoms. Choosing 2 locations from 8, there are 28 different Au arrangements, of which 4 combinations (14% of total) form superlattice structures. Gaussian noise $\mathcal{N}(0, 0.01^2)$ was added to each coordinate to augment the dataset, resulting in a total of 10,000 samples.

Sampling Three score ratios, 1:3, 1:5, and 1:7, were selected. To ensure reproducibility, the variance in the reverse diffusion process, which introduces randomness, was set to zero. Thresholds for matching rate were set with stol=0.05, angle_tol=10, and ltol=0.3.

Results The results of matching rate are shown in Table D1, and sampled structures (with a score ratio of 1:3) are shown in Figure D2. The score ratio of 1:3 exhibited the best matching rate among all. In Addition, as shown in Figure D2, the unphysical structure with overlapped atoms were obtained, which were not observed in DiffCSP. Using StructureMatcher class in pymatgen (Ong et al. 2013), we analyzed the percentage of atoms occupying the correct position in the FCC lattice. Thresholds were set at stol=0.05, angle_tol=10, and ltol=0.3. The regular arrangement rate was 95.42% for DiffCSP, while it was 90.27% with a score ratio

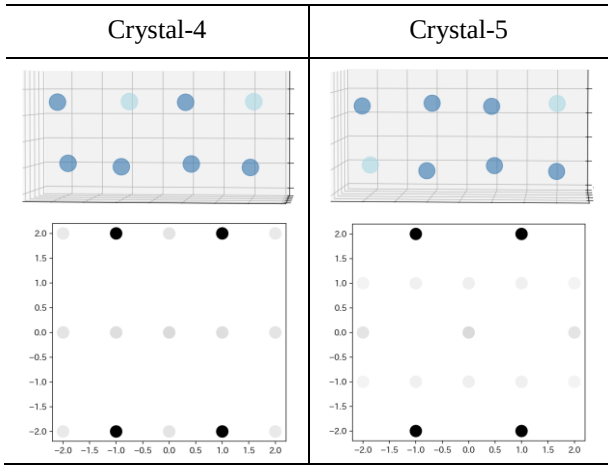

Figure D1: Crystal-4 is a superlattice structure, while Crystal-5 is not. In the superlattice structure, Au atoms are never adjacent to each other and are only surrounded by Cu atoms. The top two images show the crystal structures of each (with arrows indicating Au atoms), and the bottom two images display the X-ray diffraction intensities for each, with k = -1 shown as it highlights the differences most clearly.

Table D1: The results of the matching rates for DiffCSP and Ours (with three different ratios). Ours (1:3) indicate results from the score ratio = 1:3.

DiffCSP	**Ours (1:3)**	**Ours (1:5)**	**Ours (1:7)**
11.64%	**31.52%**	14.47%	14.21%

of 1:3, indicating more samples without regular atomic arrangement.

The highest matching rate was achieved with a score ratio of 1:3, emphasizing the importance of σ_y for better accuracy. However, in contrast to DiffCSP, the proposed method also produced invalid structure of overlapped atoms. Similar to the second experiment, the Au atoms, having more electrons and larger scattering factors than Cu atoms, exhibited greater mobility in the generation process. This likely led to cases where Au atoms moved fast, while Cu atoms stagnate, resulting in such invalid structure.

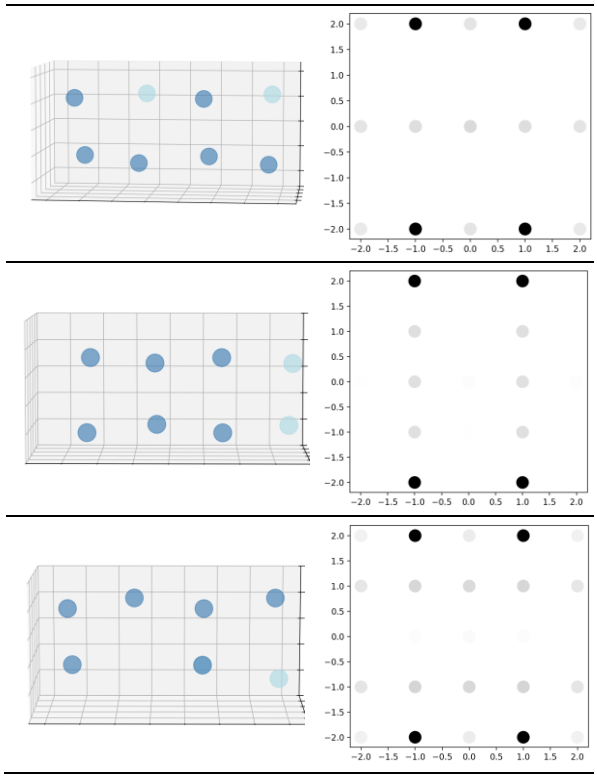

Figure D2: Sampling Results. The left images show the crystal structure obtained from the sampling results, and the right images show the corresponding X-ray diffraction intensities.