Reflection-Equivariant Diffusion for 3D Structure Determination from Isotopologue Rotational Spectra in Natural Abundance

Austin H. Cheng[1,2*] **Alston Lo**[1,2*] **Santiago Miret**[3] **Brooks H. Pate**[4] **Alán Aspuru-Guzik**[1,2]
[1]University of Toronto [2]Vector Institute [3]Intel Labs [4]University of Virginia
https://github.com/aspuru-guzik-group/kreed

Abstract

Structure determination is necessary to identify unknown organic molecules, such as those in natural products, forensic samples, the interstellar medium, and laboratory syntheses. Rotational spectroscopy enables structure determination by providing accurate 3D information about small organic molecules via their moments of inertia. Kraitchman analysis uses these moments to determine isotopic substitution coordinates, which are the unsigned $|x|, |y|, |z|$ coordinates of all atoms with natural isotopic abundance, including carbon, nitrogen, and oxygen. While unsigned substitution coordinates can verify guesses of structures, the missing $+/-$ signs make it a hard computational problem to determine the actual structure from just the substitution coordinates. To tackle this inverse problem, we develop KREED (Kraitchman REflection-Equivariant Diffusion), a diffusion generative model which infers a molecule's all-atom 3D structure conditioned on the molecular formula, moments of inertia, and unsigned substitution coordinates of carbon and other heavy atoms. KREED's top-1 predictions identify the correct 3D structure with >98% accuracy on the QM9 and GEOM datasets when provided with substitution coordinates of all heavy atoms with natural isotopic abundance. When substitution coordinates are restricted to only a subset of carbons, accuracy is retained at 91% for QM9 and 32% for GEOM. On a test set of experimentally measured substitution coordinates gathered from the literature, KREED can identify the correct all-atom 3D structure in 25 of 33 cases, demonstrating experimental applicability for context-free 3D structure determination with rotational spectroscopy.

1 Introduction

Rotational spectroscopy provides rich 3D structural information about molecules via their principal moments of inertia I_X, I_Y, I_Z (Gordy et al., 1984; Townes and Schawlow, 2013). Since a small percentage of atoms exist as isotopes in natural abundance (e.g., about 1% of carbon atoms are carbon-13), a chemical sample contains isotopically substituted versions of the molecule of interest. These isotopologues are chemically identical to the original parent molecule, but have an atom substituted with another isotope, so they have perturbed moments of inertia I_X^*, I_Y^*, I_Z^*. The change in moments caused by the isotopic substitution reveals information about the position of the substituted atom. Indeed, Kraitchman (1953) provides an expression which takes in parent and isotopologue moments and outputs the unsigned $|x|, |y|, |z|$ positions of the substituted atom in the isotopologue. The coordinates are defined in the molecule's principal axis system, where the origin is the weighted center-of-mass and the axes are the principal axes of rotation. This analysis can be repeated for

*Equal contribution. <austin@cs.toronto.edu, alston.lo@mail.utoronto.ca>

37th Conference on Neural Information Processing Systems (NeurIPS 2023) AI4Mat Workshop.

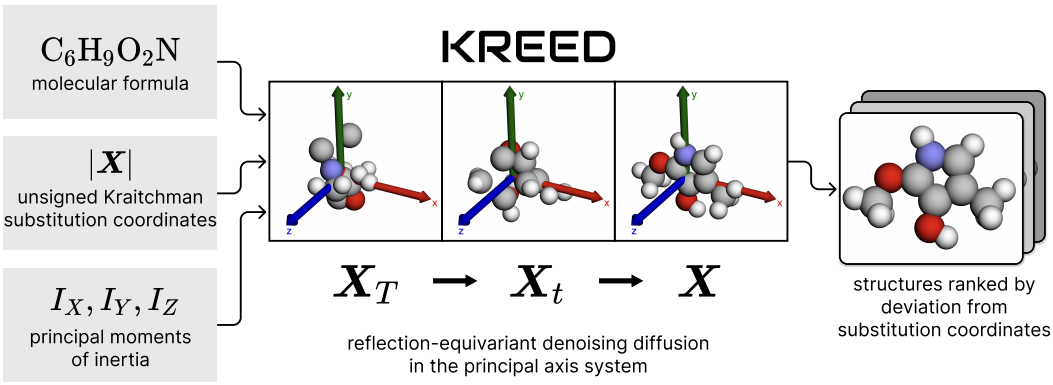

Figure 1: KREED takes as input molecular formula, Kraitchman's substitution coordinates, and principal moments of inertia *(left)* and runs a learned reverse diffusion process of Euclidean steps in the molecule's principal axis system *(center)* to obtain a ranked list of structures *(right)*.

all possible isotopologues to obtain the unsigned substitution coordinates for all atoms which have natural isotopic abundance, including carbon, nitrogen, and oxygen.

While unsigned substitution coordinates can easily verify and refine predicted 3D structures calculated from quantum chemistry (Brown et al., 2006; Shipman et al., 2011; Seifert et al., 2015), it is a difficult problem to infer the full 3D structure from just the substitution coordinates, even if the molecular formula is given. Since each atom lies in one of 8 octants, the measured substitution coordinates of a set of m atoms are consistent with 8^{m-1} possible arrangements (fix the first atom), but only one of these makes up the true 3D structure. Furthermore, substitution coordinates are not available for atoms without natural isotopic abundance (e.g., hydrogen, fluorine, and phosphorus). On top of that, zero-point vibrational effects prevent accurate measurement of substitution coordinates lying close to a principal axis.

To solve this inverse problem, we develop KREED (Kraitchman REflection-Equivariant Diffusion), an equivariant diffusion model that infers the equilibrium 3D structure of organic molecules given only the molecular formula, unsigned substitution coordinates, and principal moments of inertia (Figure 1). We define a diffusion process where all atoms take Euclidean steps in the principal axis system of the molecule and train a denoiser network to reverse this diffusion conditioned on atom types, substitution coordinates, and moments of inertia. Since the principal axis system is defined up to sign-flips, we relax the equivariance of the denoiser from E(3)- to reflection-equivariance for axially-aligned reflections across the xy, yz, xz planes. We find it is not necessary to enforce strict adherence to the substitution coordinates during the diffusion process, so generated structures are instead ranked by deviation from substitution coordinates. Because not all substitution coordinates are measurable, we perform data augmentation whereby a percentage of the substitution coordinates are randomly dropped during training.

Our approach enables context-free 3D structure determination, meaning that structure can be determined without any information of atom connectivity, SMILES string, or initial geometry. Aside from the substitution coordinates determined from rotational spectroscopy, our method requires only the molecular formula, which can be determined with high-resolution mass spectrometry (Marshall and Hendrickson, 2008). Given this information, KREED generates a ranked list of candidate all-atom 3D structures. Top-1 predictions identify the correct 3D structure with >98% accuracy on the QM9 (Ramakrishnan et al., 2014) and GEOM (Axelrod and Gomez-Bombarelli, 2022) datasets when given the substitution coordinates of all heavy atoms with natural isotopic abundance. When provided with the substitution coordinates for only carbon, and only 90% of them, top-1 predictions identify the correct 3D structure 91% of the time for QM9 and 32% of the time for GEOM. We validate KREED on a test set of experimentally measured substitution coordinates we gathered from the literature and find the model can identify the correct all-atom 3D structure in 25 of 33 cases, demonstrating potential for context-free 3D structure determination with rotational spectroscopy.

2 Background and Problem Setup

For a molecule whose 2D and 3D structure are both unknown, we wish to determine the all-atom 3D structure of its lowest energy conformer. We begin by making five assumptions: **(1)** The molecule's formula can be determined from high resolution mass spectrometry (Marshall and Hendrickson, 2008). **(2)** The molecule's structure is well-approximated as a rigid rotor, so that its rotational spectrum is entirely described by its rotational constants A, B, C. **(3)** The molecule is an asymmetric top (i.e., $A > B > C$) so that there are no molecular symmetries to exploit, which holds for the vast majority of molecules. **(4)** The molecule has a permanent dipole moment so that its rotational spectrum can be measured. Finally, **(5)** the rotational constants of the molecule and many of its naturally-abundant isotopologues can be assigned in a context-free manner without any prior guess of 3D geometry.

While quickly and reliably assigning rotational constants from a given rotational spectrum is not a solved problem, significant progress has been made. Automated fitting tools such as AUTOFIT (Seifert et al., 2015) and PGOPHER (Western, 2017; Western and Billinghurst, 2019) have increased the speed of assigning spectra, while genetic algorithms (Leo Meerts and Schmitt, 2006) and neural networks (Zaleski and Prozument, 2018) have been applied to automatically assign spectra. In particular, Yeh et al. (2019) provide an algorithm for context-free assignment of rotational spectra, which does not require prior guesses of rotational constants.

Given parent and isotopologue rotational constants, we can convert them to planar moments of inertia $P_X > P_Y > P_Z$, which are the eigenvalues of the planar dyadic (Appendix A):

$$\begin{aligned} I_X = \frac{h}{8\pi^2 A}, \quad I_Y = \frac{h}{8\pi^2 B}, \quad I_Z = \frac{h}{8\pi^2 C}, \\ P_\zeta = \frac{1}{2}(I_X + I_Y + I_Z) - I_\zeta, \quad \text{for } \zeta \in \{X, Y, Z\}, \end{aligned} \tag{1}$$

where h is Planck's constant. These moments are effective moments because measured rotational constants are averaged over the ground vibrational state, but their values are sufficiently accurate for our purposes.

Now, consider an isotopologue where a single atom of the parent molecule has been isotopically substituted. This substitution perturbs the parent molecule's mass M by some m_Δ and produces a new triplet of planar moments P_X^*, P_Y^*, P_Z^*. Kraitchman's equations (Kraitchman, 1953) relate the parent and isotopologue moments to the coordinates of the substituted atom (x, y, z), but only up to a sign:

$$|x| = \sqrt{\left(\frac{M + m_\Delta}{M m_\Delta}\right) \frac{(P_X^* - P_X)(P_Y^* - P_X)(P_Z^* - P_X)}{(P_Y - P_X)(P_Z - P_X)}}, \tag{2}$$

$$\text{with } |y| \text{ and } |z| \text{ by cyclic permutation of } X, Y, Z.$$

These equations are derived from how isotopic substitution predictably shifts and rotates the principal axes of a molecule (Figure 2, *top-left*). Since effective moments were used in Kraitchman's equations, these substitution coordinates are susceptible to errors arising from zero-point vibrational effects, so that they are systematically smaller in magnitude than the true equilibrium coordinates (Kraitchman, 1953). For coordinates lying near a principal axis, these errors can be so severe that Equation 2 produces an imaginary value. Accordingly, any unavailable substitution coordinates, either due to low natural isotopic abundance or imaginary values, are set to null values.

Finally, a matching of isotopologue rotational constants to the type of atom that was isotopically substituted is needed (e.g., determining whether a set of rotational constants comes from a ^{13}C or ^{18}O isotopologue). In the context-free setting, this matching can be determined by examining line intensities of the isotopologues and comparing to natural abundance ratios. Alternatively, if the molecular formula is not too complex, one can match isotopologues to atom types by brute force.

In summary, the available information that will be provided are **(1)** a molecular formula, **(2)** a possibly incomplete set of unsigned substitution coordinates, and **(3)** the principal planar moments of inertia of the parent molecule. In turn, we wish to obtain a ranked list of all-atom 3D structures predicted to be the true equilibrium structure.

Related work. Mayer et al. (2019) searches through heavy atom frameworks consistent with a set of substitution coordinates, and is emulated by our genetic algorithm baseline. McCarthy and Lee (2020) present a probabilistic machine learning framework for determining structures from a small set of spectroscopic parameters. Yeh et al. (2021) propose two methods for resolving sign ambiguities from Kraitchman's substitution coordinates using measurements of doubly-substituted isotopologues, electric dipole moments, and magnetic g-factors. However, as all these methods are proof-of-concept, they do not demonstrate success on large datasets.

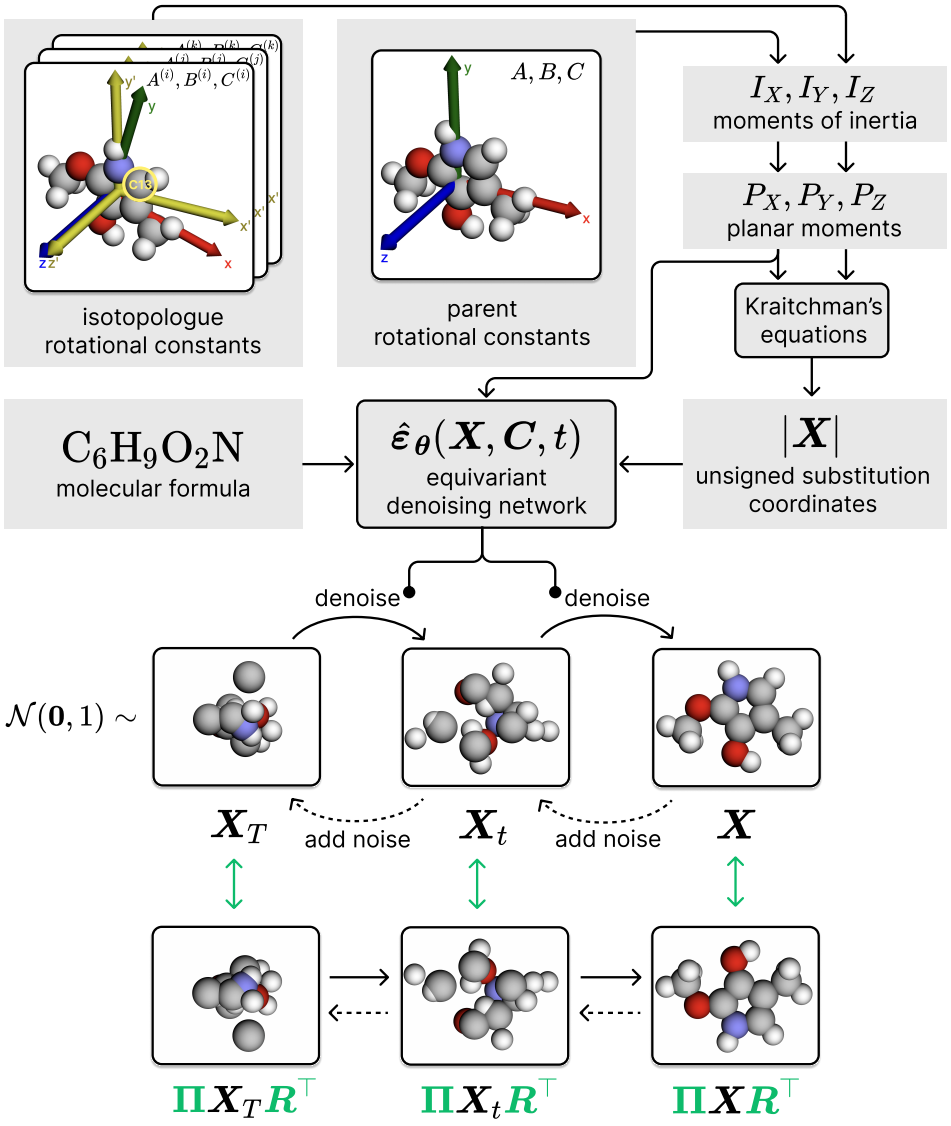

Figure 2: *Top:* Isotopologues have systematically shifted and rotated principal axes (yellow) relative to the parent's principal axes (RGB). The effect of isotopic substitution is magnified by 50× for visualization. Parent and isotopologue rotational constants are converted to planar moments of inertia and then to unsigned substitution coordinates using Kraitchman's equations. Given molecular formula, substitution coordinates, and planar moments of inertia of the parent molecule, KREED learns to denoise random point clouds into all-atom 3D structures of the molecule. *Bottom:* The denoiser model $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ is equivariant with respect to axially-aligned reflections $\boldsymbol{R}$ and node permutations $\boldsymbol{\Pi}$, so that the modelled distribution $p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{C})$ under the diffusion model is invariant to such transformations.

3 Approach

The inputs of molecular formula, substitution coordinates, and moments of inertia specify a set of incomplete constraints on the true equilibrium structure of a molecule, as many unphysical point clouds could also satisfy these conditions within a certain tolerance. However, molecules also obey several steric and electronic rules known to chemists: atoms must have satisfactory valency, bonds must be of adequate length, and bond angles must minimize strain. We leverage advances in diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) to learn and incorporate these constraints from data. Indeed, diffusion models have already demonstrated success on several 3D molecular tasks, including conformer search (Xu et al., 2022; Jing et al., 2022), docking (Corso et al., 2023), and unconditional generation (Hoogeboom et al., 2022; Schneuing et al., 2022; Vignac et al., 2023; Xu et al., 2023). The basic idea behind diffusion models is that the true data distribution $p_{\text{data}}(\boldsymbol{X}|\boldsymbol{C})$ can be transformed into a standard Gaussian distribution by progressively adding Gaussian noise across multiple timesteps t. By training a neural network $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ to predict the reverse of these diffusion steps, random noise can be iteratively denoised into samples from the original data distribution. We recall the standard formulation of a diffusion model in Appendix B.

For an unknown N-atom molecule, let $\boldsymbol{C}$ be a node feature matrix encoding the prior information that is available. Concretely, we concatenate the following features of the molecule:

$$\boldsymbol{C} = \left(\boldsymbol{a} \;\middle|\; \boldsymbol{m} \;\middle|\; |\boldsymbol{X}| \;\middle|\; \boldsymbol{S} \;\middle|\; P_X, P_Y, P_Z \right) \in \mathbb{R}^{N \times 11}, \tag{3}$$

where $\boldsymbol{a}$ are its atomic numbers, $\boldsymbol{m}$ are its atomic masses, $|\boldsymbol{X}|$ are its unsigned substitution coordinates (or 0 where not given), $\boldsymbol{S}$ is a binary mask that indicates which elements of $|\boldsymbol{X}|$ are missing, and P_X, P_Y, P_Z are its planar moments of inertia for the parent isotopologue (repeated along each row). Then we aim to learn the true distribution of 3D conformations $\boldsymbol{X} \in \mathbb{R}^{N \times 3}$ conditioned on $\boldsymbol{C}$ with a diffusion model $p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{C})$.

3.1 Diffusion in the principal axis system

Molecules in 3D contain geometric symmetries under translations, rotations, and reordering of atoms, which our diffusion model should account for to ensure good generalization (Elesedy and Zaidi, 2021). A natural approach is to consider the principal axis system of the molecule, which is the coordinate system whose origin is the mass-weighted center-of-mass (CoM) and whose axes are the principal axes of rotation, or the orthonormal eigenvectors of the inertia matrix (Appendix A). This choice is motivated by the substitution coordinates being given in the principal axis system. In particular, fixing the origin to the CoM removes translational symmetries (Xu et al., 2022), while aligning the molecule to its principal axes reduces symmetries under rotoreflections to axially-aligned reflections across the xy, yz, xz planes. These reflectional symmetries persist since if $\boldsymbol{v}$ is an eigenvector of the inertia matrix then so is $-\boldsymbol{v}$, so the principal axes are only unique up to sign flips. We then model a reflection-invariant distribution using a reflection-equivariant denoiser (Figure 2, *bottom*).

3.1.1 Zero CoM subspace

Fixing the CoM to the origin amounts to performing the diffusion process in the $3(N-1)$-dimensional linear subspace

$$\mathbb{U} = \left\{ \boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top \in \mathbb{R}^{N \times 3} \;\middle|\; \frac{1}{\sum_{j=1}^{N} m_j} \sum_{i=1}^{N} m_i \boldsymbol{x}_i = \boldsymbol{0} \right\}. \tag{4}$$

Unlike $\mathbb{R}^{N \times 3}$, points in $\mathbb{U}$ cannot be superimposed by translations. To diffuse over $\mathbb{U}$, we require two minor changes from a regular diffusion model on $\mathbb{R}^{N \times 3}$ (Appendix B): **(1)** whenever a sample is drawn from a Gaussian distribution, it must be orthogonally projected onto $\mathbb{U}$, and **(2)** the output of the network $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ is restricted to $\mathbb{U}$. This approach is similar to previous works which fix the *unweighted* CoM at $\boldsymbol{0}$ (Xu et al., 2022; Hoogeboom et al., 2022), with the added difference that an orthogonal projection onto $\mathbb{U}$ is not equivalent to simply removing the weighted CoM. Appendix Algorithms 1 and 2 summarize the modified procedures for training and sampling. Appendix C.1 formalizes and proves these claims.

3.1.2 Axially-aligned reflection invariance

Once the CoM has been fixed to the origin, diffusion takes place in the principal axis system of the true molecule by construction. The remaining symmetries are stated as follows. For all axially-aligned reflections $\boldsymbol{R} \in \{\mathrm{diag}(\boldsymbol{b}) \mid \boldsymbol{b} \in \{-1, +1\}^3\}$ and node permutations $\boldsymbol{\Pi}$ that map $\mathbb{U}$ onto itself, we desire that:

$$p_{\boldsymbol{\theta}}(\boldsymbol{\Pi X R}^\top | \boldsymbol{\Pi C}) = p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{C}). \tag{5}$$

That is, the modelled distribution is invariant to axially-aligned reflections of $\boldsymbol{X}$ and simultaneous reorderings of $\boldsymbol{X}$ and $\boldsymbol{C}$. This is satisfied if the denoiser network $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ is correspondingly equivariant (Appendix C.2):

$$\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\boldsymbol{\Pi X R}^\top, \boldsymbol{\Pi C}, t) = \boldsymbol{\Pi}\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{C}, t)\boldsymbol{R}^\top. \tag{6}$$

To ensure this, we implement $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ with an architecture inspired from E(n)-equivariant graph neural networks (EGNNs) (Satorras et al., 2021). In the $n = 3$ case, EGNNs are designed to be equivariant under the E(3) group, which is the group of rigid 3D motions consisting of translations, reflections, rotations, and combinations thereof. By modifying the EGNN with edge features that are reflection- but not E(3)-invariant, we relax its E(3)- to reflection-equivariance. We also found that incorporating Transformer-like elements (Vaswani et al., 2017) improved training dynamics and stability. Further architectural details are given in Appendix E.2.

Frame averaging (Puny et al., 2022; Duval et al., 2023) arises as an unexpected connection to our approach. This method ensures E(3)-equivariance by averaging over the frame of the E(3) group, where the frame consists of the 2^3 possible point clouds that are aligned to the principal components of the point clouds and have the same *unweighted* CoM. However, instead of aligning to principal components, we align to principal axes of rotation and fix the *weighted* CoM to be zero.

3.2 Training and inference

During training, we process all examples so that they sit in their principal axis system. Simultaneously, we can easily compute the moments of inertia of the molecule and compute unsigned substitution coordinates by discarding signs (Appendix Algorithm 3). We only use substitution coordinates of atoms with naturally abundant isotopes, which for our datasets includes B, C, N, O, Si, S, Cl, Br, Hg and excludes H, F, Al, P, As, I, Bi. However, not all of these substitution coordinates are available in practice due to low natural isotopic abundance or zero-point vibrational effects. To mimic this effect, we further apply a random dropout with probability p on the substitution coordinates, with p itself being uniformly sampled from an interval $[p_{\text{min}}, p_{\text{max}}] \subseteq [0, 1]$ for each training example. At inference time, we sample K predictions for each example and rank predictions by deviation of their unsigned coordinates to the original unsigned substitution coordinates. Additional details on training and sampling, including an explicit inference workflow, are available in Appendix E.3.

4 Experiments

Baseline. To determine the effectiveness of search-based methods for solving this problem, we develop a genetic algorithm that searches over heavy atom frameworks by finding binary $+/-$ signs for heavy atoms with specified unsigned substitution coordinates and continuous positions for heavy atoms without. The search is guided by a fitness function which measures agreement with zero CoM and the provided moments, combined with a likelihood term that is simply a pair distribution function of heavy atoms in the training set, inspired by Mayer et al. (2019). Hydrogens are added afterwards using Hydride (Kunzmann et al., 2022). More details are available in Appendix D.

Datasets. QM9 is an enumeration of 134k single-conformer molecules containing C, H, O, N, F with up to 9 heavy atoms calculated at the B3LYP/6-31G(2df,p) level of theory. GEOM is a dataset of 292k drug-like molecules with conformers calculated by CREST (Pracht et al., 2020) at a semiempirical extended tight-binding level GFN2-xTB (Bannwarth et al., 2019). For GEOM, we use only the 30 lowest-energy conformers for each molecule, totalling 6.9M conformers. Each dataset is then partitioned into training, validation, and test sets using an 80:10:10% random split by molecule.

To provide a more realistic benchmark, we also drop substitution coordinates of all non-carbon atoms, followed by dropping 10% of carbon substitution coordinates. This emulates the fact that non-carbon isotopologues are sometimes difficult to detect in a spectrum due to low but non-zero abundance. For

Method	Task	Correctness (%) $k=1$	Correctness (%) $k=5$	Median RMSD (Å) $k=1$	Median RMSD (Å) $k=5$
Genetic algorithm	QM9	7.33	12.4	0.842	0.546
	QM9-C	0.127	0.225	1.29	1.05
	GEOM	0.0308	0.0377	2.05	1.77
	GEOM-C	0.00342	0.00685	2.31	2.08
KREED	QM9	99.9	99.9	0.00626	0.00625
	QM9-C	91.3	93.1	0.00935	0.00882
	GEOM	98.9	99.2	0.00617	0.00616
	GEOM-C	32.6	35.8	1.17	0.765

Table 1: The genetic algorithm and KREED are benchmarked on the test sets of QM9 and GEOM. Performance is measured by connectivity correctness and median all-atom RMSD.

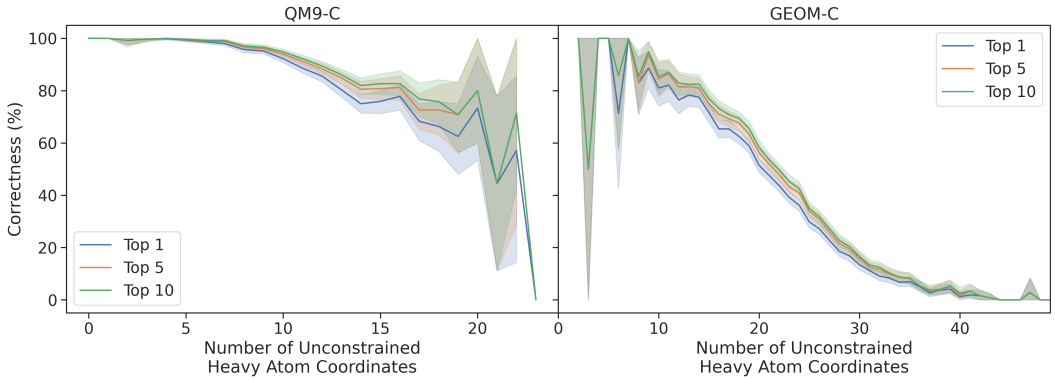

Figure 3: Average connectivity correctness of KREED on QM9-C and GEOM-C for samples with different numbers of unconstrained heavy atom coordinates. Shading shows the 95% confidence interval. The fewer the number of substitution coordinates that are provided, the more the number of unconstrained heavy atom coordinates, and the more difficult the task, as shown by decreases in average correctness.

example, the isotopic abundances of nitrogen and oxygen are respectively 2.5× and 5× lower than that for carbon. We label these tasks as QM9-C and GEOM-C, both of which drop approximately 34% of the original naturally-abundant isotopic substitution coordinates.

One model was trained on QM9 and tested on the QM9 and QM9-C tasks, and another model was trained on GEOM and tested on the GEOM and GEOM-C tasks. For the test set of GEOM, only predictions for the lowest-energy conformer were evaluated, as the lowest-energy conformer will have the highest population in experiments. For each test example, $K = 10$ samples were generated and ranked by deviation from unsigned substitution coordinates.

Metrics. We evaluate generated samples in terms of connectivity correctness and all-atom RMSD. A prediction is connectivity correct if RDKit's xyz2mol `rdkit.Chem.rdDetermineBonds` (Kim and Kim, 2015) returns the same SMILES string for both the prediction and the ground truth. Connectivity correctness implies that the prediction and ground truth have the same bond connectivity, but may not necessarily be the same enantiomer or diastereomer. We additionally measure each prediction's all-atom RMSD after alignment to the ground truth via minimum RMSD over all 2^3 possible reflections. Since our diffusion model is permutation invariant, node orderings are not preserved, and we must find an assignment of each atom in the ground truth to each atom in the predicted sample. To do so, for each reflection, we solve a linear assignment problem (Crouse, 2016) to match atoms that are close in space and have the same atom type but may not have the same node index.

Accuracy. Table 1 indicates that KREED can predict structures with near perfect accuracy on both QM9 and GEOM when provided with all naturally abundant substitution coordinates, even without

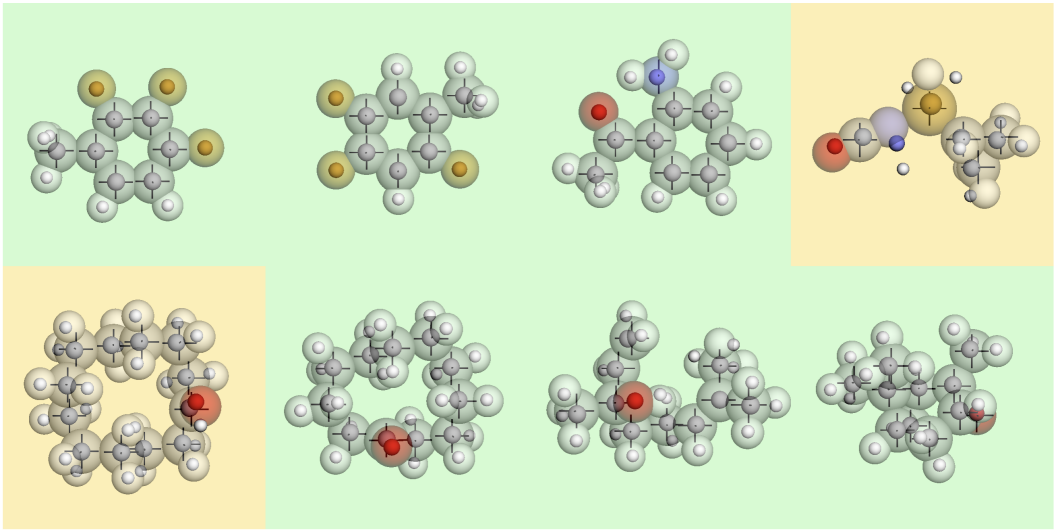

Figure 4: Top-1 predictions for experimental substitution coordinates of molecules which do not appear in QM9 or GEOM. The transparent structure is the ground truth, while the small spheres indicate the top-1 predicted structure. Green indicates all-atom correctness while yellow indicates heavy atom correctness. The presence of black pins indicates whether a substitution coordinate in that direction was available.

hydrogen, fluorine, and phosphorus. When restricting the provided substitution coordinates, the model retains 91.3% top-1 accuracy for QM9-C, while performance drops to 32.6% top-1 accuracy for GEOM-C. In comparison, performance by the genetic algorithm is poor. While this simple baseline demonstrates appreciable results for QM9, even achieving a top-10 heavy-atom-correctness of 46.4% (Appendix Table 3), it is not able to predict structures from GEOM. Low baseline performance is likely due to inefficient generation of continuous coordinates and a poorly adapted fitness function for larger molecules. Additional top-k metrics and visualized top-1 predictions for all methods and tasks are shown in Appendix F.

The significant drop in performance when moving from GEOM to GEOM-C is in line with the fact that GEOM contains molecules with significantly more atoms than QM9: QM9 contains an average of 8.8 heavy atoms per molecule, while GEOM contains an average of 24.8 heavy atoms per molecule. In Figure 3, we see that the difficulty of an example is correlated with the number of unconstrained heavy atom coordinates, which is the number of heavy atom coordinates for which substitution coordinates are *not* provided. Indeed, further experiments found that even if provided with *only* molecular formula and moments, KREED can still obtain reasonable accuracy on QM9 (48.8% top-100 accuracy), owing to the small size of molecules in QM9. We discuss these results in Appendix F.1.

4.1 Experimentally measured substitution coordinates

To benchmark KREED's applicability to experimental data, we extracted from the literature a small dataset of 33 conformers with experimentally measured substitution coordinates. Experimental measurements are subject to zero-point vibrational effects, which cause small inaccuracies in substitution coordinates. For these experiments, we generated $K = 100$ samples for each test example. For the 8 examples which do not appear in QM9 or GEOM, our top-1 predictions are shown in Figure 4. KREED can predict the correct heavy atom frameworks for all examples and the correct all-atom structures for 6 of 8 examples, demonstrating strong performance even on examples that are both out-of-dataset and subject to zero-point vibrational effects.

Of the remaining molecules, 7 were very similar to examples in QM9 or GEOM, while 18 did not have ground truth Cartesian coordinates provided, so they were manually verified by visual comparison to pictures in the original paper. In total, 25 of the 33 examples were all-atom correct, while 29 of the 33 examples were correct up to hydrogens. The robustness of KREED to inaccuracies due to zero-point

vibrational effects may be attributed to its loose conditioning on substitution coordinates; since the model is not forced to have exact agreement with the substitution coordinates, it can account for errors in them. Top-1 predictions are visualized in Appendix G along with references.

5 Conclusion

We introduce KREED, a reflection-equivariant diffusion model for inferring all-atom 3D structures from molecular formulae, Kraitchman's substitution coordinates, and moments of inertia. We validate our approach on large datasets and obtain high accuracy, especially as more substitution coordinates are provided. Additionally, we find that KREED is applicable to experimentally measured substitution coordinates, demonstrating its potential for context-free 3D structure determination.

Our approach relies on being able to determine at least most of the substitution coordinates accurately. The main obstacle to this is being able to detect and assign rotational constants to enough isotopologues, even for atoms with low abundance such as oxygen and nitrogen. We must also know what type of atom was substituted in each isotopologue, which may be nontrivial to resolve. Furthermore, our method inherits the constraints of Kraitchman analysis and rotational spectroscopy, such as the rigid rotor approximation, requirement of a permanent dipole moment, and inability to distinguish enantiomers. Lastly, we require a second instrument to determine molecular formula.

However, we believe that many of these limitations can be addressed in future work. One promising direction is to explore more powerful methods for conditioning the diffusion model, such as guidance (Dhariwal and Nichol, 2021; Ho and Salimans, 2022). Given that analytic expressions for moments of inertia are available, it would also be interesting if the diffusion process could be constrained so that the moments of inertia are exactly preserved by the end of the diffusion process. This would remove the need for substitution coordinates and enable structure determination solely from moments and molecular formula. One might be able to sidestep these input requirements altogether by conditioning directly on rotational spectra. An orthogonal direction could be to evaluate reflection-equivariant diffusion in the principal axis system as a general method for modelling 3D point cloud data.

Acknowledgments

We thank Naruki Yoshikawa, Luca Thiede, and Kin Long Kelvin Lee for helpful discussions, and Andy Cai for help with visualizations. Resources used in preparing this research were provided by Intel, the Province of Ontario, the Government of Canada through CIFAR, companies sponsoring the Vector Institute (`www.vectorinstitute.ai/#partners`), Calcul Québec, and the Digital Research Alliance of Canada (`alliancecan.ca`). This research was undertaken thanks in part to funding provided to the University of Toronto's Acceleration Consortium from the Canada First Research Excellence Fund CFREF-2022-00042. We acknowledge the Defense Advanced Research Projects Agency (DARPA) under the Accelerated Molecular Discovery Program under Cooperative Agreement No. HR00-11920027 dated August 1, 2019. A.A.-G. thanks Anders G. Frøseth for his generous support. A.A.-G also acknowledges the generous support of Natural Resources Canada and the Canada 150 Research Chairs program.

We acknowledge the Python community (Van Rossum et al., 1995; Oliphant, 2007) for developing the core set of tools that enabled this work, including PyTorch (Paszke et al., 2019), PyTorch Lightning (Falcon and The PyTorch Lightning team, 2019), DGL (Wang et al., 2019), RDKit (Landrum et al., 2023), py3Dmol (Rego and Koes, 2015), Jupyter (Kluyver et al., 2016), Matplotlib (Hunter, 2007), seaborn (Waskom, 2021), NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020), and pandas (The pandas development team).

References

Walter Gordy, Robert L Cook, and Arnold Weissberger. *Microwave molecular spectra*, volume 18. Wiley New York, 1984. URL `https://app.knovel.com/kn/resources/kpMMSE0001/toc`.

Charles H Townes and Arthur L Schawlow. *Microwave spectroscopy*. Courier Corporation, 2013.

J Kraitchman. Determination of molecular structure from microwave spectroscopic data. *American Journal of Physics*, 21(1):17–24, 1953.

Gordon G Brown, Brian C Dian, Kevin O Douglass, Scott M Geyer, and Brooks H Pate. The rotational spectrum of epifluorohydrin measured by chirped-pulse Fourier transform microwave spectroscopy. *Journal of Molecular Spectroscopy*, 238(2):200–212, 2006.

Steven T Shipman, Justin L Neill, Richard D Suenram, Matt T Muckle, and Brooks H Pate. Structure determination of strawberry aldehyde by broadband microwave spectroscopy: Conformational stabilization by dispersive interactions. *The Journal of Physical Chemistry Letters*, 2(5):443–448, 2011.

Nathan A Seifert, Ian A Finneran, Cristobal Perez, Daniel P Zaleski, Justin L Neill, Amanda L Steber, Richard D Suenram, Alberto Lesarri, Steven T Shipman, and Brooks H Pate. AUTOFIT, an automated fitting tool for broadband rotational spectra, and applications to 1-hexanal. *Journal of Molecular Spectroscopy*, 312:13–21, 2015.

Alan G Marshall and Christopher L Hendrickson. High-resolution mass spectrometers. *Annu. Rev. Anal. Chem.*, 1:579–599, 2008.

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

Simon Axelrod and Rafael Gomez-Bombarelli. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):1–14, 2022.

Colin M Western. PGOPHER: A program for simulating rotational, vibrational and electronic spectra. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 186:221–242, 2017.

Colin M Western and Brant E Billinghurst. Automatic and semi-automatic assignment and fitting of spectra with PGOPHER. *Physical Chemistry Chemical Physics*, 21(26):13986–13999, 2019.

W Leo Meerts and Michael Schmitt. Application of genetic algorithms in automated assignments of high-resolution spectra. *International Reviews in Physical Chemistry*, 25(3):353–406, 2006.

Daniel P Zaleski and Kirill Prozument. Automated assignment of rotational spectra using artificial neural networks. *The Journal of chemical physics*, 149(10):104106, 2018.

Lia Yeh, Lincoln Satterthwaite, and David Patterson. Automated, context-free assignment of asymmetric rotor microwave spectra. *The Journal of chemical physics*, 150(20):204122, 2019.

Kevin J Mayer, Brooks Pate, Eleanor Patrinely, Emmit Pert, Austin Cheng, Kira Baugh, Kevyn Hadley, Sean Lee, George Fairman, Jake Butler, et al. The feasibility of determining the carbon framework geometry of a molecule from analysis of the carbon-13 isotopologue rotational spectra in natural abundance. In *74th International Symposium on Molecular Spectroscopy*, 2019.

Michael McCarthy and Kin Long Kelvin Lee. Molecule identification with rotational spectroscopy and probabilistic deep learning. *The Journal of Physical Chemistry A*, 124(15):3002–3017, 2020.

Lia Yeh, Dylan Finestone, Lincoln Satterthwaite, Jieyu Yan, and David Patterson. Progress made towards context-free molecular structure determination from isotopologue rotational spectroscopy. In *2021 International Symposium on Molecular Spectroscopy*, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

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

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi S. Jaakkola. Torsional diffusion for molecular conformer generation. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi S. Jaakkola. DiffDock: Diffusion steps, twists, and turns for molecular docking. In *The Eleventh International Conference on Learning Representations*, 2023.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3D. In *International Conference on Machine Learning*, pages 8867–8887. PMLR, 2022.

Arne Schneuing, Yuanqi Du, Charles Harris, Arian Jamasb, Ilia Igashov, Weitao Du, Tom Blundell, Pietro Lió, Carla Gomes, Max Welling, et al. Structure-based drug design with equivariant diffusion models. *arXiv preprint arXiv:2210.13695*, 2022.

Clement Vignac, Nagham Osman, Laura Toni, and Pascal Frossard. Midi: Mixed graph and 3d denoising diffusion for molecule generation. In *ICLR 2023 - Machine Learning for Drug Discovery workshop*, 2023.

Minkai Xu, Alexander S Powers, Ron O Dror, Stefano Ermon, and Jure Leskovec. Geometric latent diffusion models for 3D molecule generation. In *International Conference on Machine Learning*, pages 38592–38610. PMLR, 2023.

Bryn Elesedy and Sheheryar Zaidi. Provably strict generalisation benefit for equivariant models. In *International Conference on Machine Learning*, pages 2959–2969. PMLR, 2021.

Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9323–9332. PMLR, 18–24 Jul 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Omri Puny, Matan Atzmon, Edward J. Smith, Ishan Misra, Aditya Grover, Heli Ben-Hamu, and Yaron Lipman. Frame averaging for invariant and equivariant network design. In *International Conference on Learning Representations*, 2022.

Alexandre Agm Duval, Victor Schmidt, Alex Hernández-García, Santiago Miret, Fragkiskos D Malliaros, Yoshua Bengio, and David Rolnick. Faenet: Frame averaging equivariant gnn for materials modeling. In *International Conference on Machine Learning*, pages 9013–9033. PMLR, 2023.

Patrick Kunzmann, Jacob Marcel Anter, and Kay Hamacher. Adding hydrogen atoms to molecular models via fragment superimposition. *Algorithms for Molecular Biology*, 17(1):1–8, 2022.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Physical Chemistry Chemical Physics*, 22(14):7169–7192, 2020.

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. GFN2-xTB - An accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of chemical theory and computation*, 15 (3):1652–1671, 2019.

Yeonjoon Kim and Woo Youn Kim. Universal structure conversion method for organic molecules: from atomic connectivity to three-dimensional geometry. *Bulletin of the Korean Chemical Society*, 36(7):1769–1777, 2015.

David F Crouse. On implementing 2D rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, 2016.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Guido Van Rossum, Fred L Drake, et al. *Python reference manual*, volume 111. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

Travis E Oliphant. Python for scientific computing. *Computing in science & engineering*, 9(3):10–20, 2007.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

William Falcon and The PyTorch Lightning team. PyTorch Lightning, March 2019. URL https://github.com/Lightning-AI/lightning.

Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019.

Greg Landrum, Paolo Tosco, Brian Kelley, Ric, David Cosgrove, sriniker, gedeck, Riccardo Vianello, NadineSchneider, Eisuke Kawashima, Dan N, Gareth Jones, Andrew Dalke, Brian Cole, Matt Swain, Samo Turk, AlexanderSavelyev, Alain Vaucher, Maciej Wójcikowski, Ichiru Take, Daniel Probst, Kazuya Ujihara, Vincent F. Scalfani, guillaume godin, Juuso Lehtivarjo, Axel Pahl, Rachel Walker, Francois Berenger, jasondbiggs, and strets123. rdkit/rdkit: 2023_09_1 (q3 2023) release beta, October 2023. URL https://doi.org/10.5281/zenodo.8413907.

Nicholas Rego and David Koes. 3Dmol.js: Molecular visualization with WebGL. *Bioinformatics*, 31 (8):1322–1324, 2015.

Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica B Hamrick, Jason Grout, Sylvain Corlay, et al. Jupyter notebooks-a publishing format for reproducible computational workflows. *Elpub*, 2016:87–90, 2016.

John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(03): 90–95, 2007.

Michael L Waskom. Seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60): 3021, 2021.

Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020. doi: 10.1038/s41586-020-2649-2.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2. URL https://doi.org/10.1038/s41592-019-0686-2.

The pandas development team. pandas-dev/pandas: Pandas. URL https://github.com/pandas-dev/pandas.

Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

Shufang Xie, Huishuai Zhang, Junliang Guo, Xu Tan, Jiang Bian, Hany Hassan Awadalla, Arul Menezes, Tao Qin, and Rui Yan. Residual: Transformer with dual residual connections. *arXiv preprint arXiv:2304.14802*, 2023.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, et al. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*, 2022.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Jerry Ma and Denis Yarats. On the adequacy of untuned warmup for adaptive optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8828–8836, May 2021.

KP Rajappan Nair, Sven Herbers, Daniel A Obenchain, and Jens-Uwe Grabow. Internal methyl rotation and molecular structure of trifluorotoluenes: Microwave rotational spectra of 2, 3, 4-and 2, 4, 5-trifluorotoluene. *Canadian Journal of Physics*, 98(6):543–550, 2020.

Giovanna Salvitti, Susana Blanco, Juan Carlos Lopez, S Melandri, L Evangelisti, and A Maris. Probing intra-and inter-molecular interactions through rotational spectroscopy: The case of the odorant 2'-aminoacetophenone and its 1:1 water and neon complexes. *The Journal of Chemical Physics*, 157(14), 2022.

Gamil A Guirgis, Sahand M Askarian, Tamia Morris, Michael H Palmer, Brooks H Pate, and Nathan A Seifert. Molecular structure of cyclopropyl (isocyanato) silane: A combined microwave spectral and theoretical study. *The Journal of Physical Chemistry A*, 119(49):11875–11881, 2015.

Ecaterina Burevschi and M Eugenia Sanz. Seven conformations of the macrocycle cyclododecanone unveiled by microwave spectroscopy. *Molecules*, 26(17):5162, 2021.

Valerie WY Tsoi, Ecaterina Burevschi, Shefali Saxena, and M Eugenia Sanz. Conformational panorama of cycloundecanone: A rotational spectroscopy study. *The Journal of Physical Chemistry A*, 126(36):6185–6193, 2022.

María Mar Quesada-Moreno, Anna Krin, and Melanie Schnell. Analysis of thyme essential oils using gas-phase broadband rotational spectroscopy. *Physical Chemistry Chemical Physics*, 21(48): 26569–26579, 2019.

EM Neeman, N Osseiran, and TR Huet. The gas-phase structure determination of α-pinene oxide: An endo-cyclic epoxide of atmospheric interest. *The Journal of Chemical Physics*, 158(15), 2023.

Ha Vinh Lam Nguyen. The heavy atom substitution and semi-experimental equilibrium structures of 2-ethylfuran obtained by microwave spectroscopy. *Journal of Molecular Structure*, 1208:127909, 2020.

Cristóbal Pérez, Elena Caballero-Mancebo, Alberto Lesarri, Emilio J Cocinero, Ibon Alkorta, Richard D Suenram, Jens-Uwe Grabow, and Brooks H Pate. The conformational map of volatile anesthetics: Enflurane revisited. *Chemistry–A European Journal*, 22(28):9804–9811, 2016.

Frank E Marshall, Galen Sedo, Channing West, Brooks H Pate, Stephanie M Allpress, Corey J Evans, Peter D Godfrey, Don McNaughton, and GS Grubbs II. The rotational spectrum and complete heavy atom structure of the chiral molecule verbenone. *Journal of Molecular Spectroscopy*, 342: 109–115, 2017.

Sabrina Zinn and Melanie Schnell. Flexibility at the fringes: Conformations of the steroid hormone β-estradiol. *ChemPhysChem*, 19(21):2915–2920, 2018.

Christina Dindic, Arne Lüchow, Natalja Vogt, Jean Demaison, and Ha Vinh Lam Nguyen. Equilibrium structure in the presence of methyl internal rotation: Microwave spectroscopy and quantum chemistry study of the two conformers of 2-acetylfuran. *The Journal of Physical Chemistry A*, 125 (23):4986–4997, 2021.

Vinh Van, Wolfgang Stahl, and Ha Vinh Lam Nguyen. The heavy atom microwave structure of 2-methyltetrahydrofuran. *Journal of Molecular Structure*, 1123:24–29, 2016.

Mariyam Fatima, Cristóbal Pérez, Benjamin E Arenas, Melanie Schnell, and Amanda L Steber. Benchmarking a new segmented K-band chirped-pulse microwave spectrometer and its application to the conformationally rich amino alcohol isoleucinol. *Physical Chemistry Chemical Physics*, 22 (30):17042–17051, 2020.

Camilla Calabrese, Iciar Uriarte, Aran Insausti, Montserrat Vallejo-López, Francisco J Basterretxea, Stephen A Cochrane, Benjamin G Davis, Francisco Corzana, and Emilio J Cocinero. Observation of the unbiased conformers of putative DNA-scaffold ribosugars. *ACS Central Science*, 6(2): 293–303, 2020.

Arsh S Hazrah, Sadisha Nanayakkara, Nathan A Seifert, Elfi Kraka, and Wolfgang Jäger. Structural study of 1-and 2-naphthol: New insights into the non-covalent H–H interaction in cis-1-naphthol. *Physical Chemistry Chemical Physics*, 24(6):3722–3732, 2022.

Keisuke Ohba, Tsuyoshi Usami, Yoshiyuki Kawashima, and Eizi Hirota. Fourier transform microwave spectra and ab initio calculation of N-ethylformamide. *Journal of molecular structure*, 744:815–819, 2005.

Sabrina Zinn, Thomas Betz, Chris Medcraft, and Melanie Schnell. Structure determination of trans-cinnamaldehyde by broadband microwave spectroscopy. *Physical Chemistry Chemical Physics*, 17 (24):16080–16085, 2015.

Juncheng Lei, Jiaqi Zhang, Gang Feng, Jens-Uwe Grabow, and Qian Gou. Conformational preference determined by inequivalent n-pairs: Rotational studies on acetophenone and its monohydrate. *Physical Chemistry Chemical Physics*, 21(41):22888–22894, 2019.

Ryan G Bird, Vanesa Vaquero-Vara, Daniel P Zaleski, Brooks H Pate, and David W Pratt. Chirped-pulsed FTMW spectra of valeric acid, 5-aminovaleric acid, and δ-valerolactam: A study of amino acid mimics in the gas phase. *Journal of Molecular Spectroscopy*, 280:42–46, 2012.

Meng Li, Yang Zheng, Jiayi Li, Jens-Uwe Grabow, Xuefang Xu, and Qian Gou. Aqueous microsolvation of 4-hydroxy-2-butanone: Competition between intra-and inter-molecular hydrogen bonds. *Physical Chemistry Chemical Physics*, 24(33):19919–19926, 2022.

A Moments of inertia

For an N-atom molecule, let m_i and $\boldsymbol{x}_i \in \mathbb{R}^3$ be the mass and position of the i-th atom. Assume also that the coordinates have been centered to have zero CoM, i.e., $\sum_{i=1}^N m_i \boldsymbol{x}_i = \mathbf{0}$. Then the inertia matrix is given by

$$\mathbf{I} = \sum_{i=1}^{N} m_i \left(\|\boldsymbol{x}_i\|_2^2 \boldsymbol{I}_3 - \boldsymbol{x}_i \boldsymbol{x}_i^\top \right) \in \mathbb{R}^{3\times 3}, \tag{7}$$

where $\boldsymbol{I}_3$ is the identity matrix. The eigenvalues I_X, I_Y, I_Z of $\mathbf{I}$ are the principal moments of inertia, while its orthonormal eigenvectors form the principal axes of rotation. A closely related matrix is the planar dyadic

$$\mathbf{P} = \sum_{i=1}^{N} m_i \boldsymbol{x}_i \boldsymbol{x}_i^\top \in \mathbb{R}^{3\times 3}, \tag{8}$$

which contains the same information but leads to simpler expressions for Kraitchman's equations. The planar dyadic shares the same eigenvectors as the inertia matrix, and its eigenvalues P_X, P_Y, P_Z are the planar moments of inertia, which are related to I_X, I_Y, I_Z by Equation 1.

B Denoising Diffusion

At a high level, diffusion models learn to iteratively *denoise* samples drawn from an elementary prior distribution into samples from the desired distribution. Given a real data sample $\boldsymbol{x} \in \mathbb{R}^m$, let $\boldsymbol{z}_0, \ldots, \boldsymbol{z}_T$ be a sequence of increasingly noisier random variables drawn from the isotropic Gaussian distributions

$$q(\boldsymbol{z}_t|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}_t; \alpha_t \boldsymbol{x}, \sigma_t^2), \tag{9}$$

where $\alpha_t \in (0,1)$ are chosen to be decreasing with $\alpha_0 \approx 1$ and $\alpha_T \approx 0$, and $\sigma_t^2 = 1 - \alpha_t^2$. We also require α_T to be sufficiently small such that $\boldsymbol{z}_T$ is nearly indistinguishable from random noise, i.e., $q(\boldsymbol{z}_T|\boldsymbol{x}) \approx \mathcal{N}(\boldsymbol{z}_T; \mathbf{0}, 1)$. Now, we can equivalently model the variables through a Markovian noising process with isotropic Gaussian transitions

$$q(\boldsymbol{z}_t|\boldsymbol{z}_{t-1}) = \mathcal{N}\left(\boldsymbol{z}_t; \tilde{\alpha}_t \boldsymbol{z}_{t-1}, \tilde{\sigma}_t^2\right), \tag{10}$$

where $\tilde{\alpha}_t = \alpha_t / \alpha_{t-1}$ and $\tilde{\sigma}_t^2 = \sigma_t^2 - \tilde{\alpha}_t^2 \sigma_{t-1}^2$, and $\alpha_{-1} = 1$ and $\boldsymbol{z}_{-1} = \boldsymbol{x}$ for the case $t = 0$. In fact, the transition posteriors conditioned on $\boldsymbol{x}$ are again isotropic and Gaussian: for $t \geq 1$,

$$q(\boldsymbol{z}_{t-1}|\boldsymbol{x}, \boldsymbol{z}_t) = \mathcal{N}\left(\boldsymbol{z}_{t-1}; \frac{\tilde{\alpha}_t \sigma_{t-1}^2}{\sigma_t^2} \boldsymbol{z}_t + \frac{\alpha_{t-1} \tilde{\sigma}_t^2}{\sigma_t^2} \boldsymbol{x}, \frac{\tilde{\sigma}_t^2 \sigma_{t-1}^2}{\sigma_t^2}\right). \tag{11}$$

Equation 11 specifies a true denoising process in which we can transform a heavily corrupted sample $\boldsymbol{z}_T \sim q(\boldsymbol{z}_T|\boldsymbol{x})$ back into its original state $\boldsymbol{x}$.

However, $\boldsymbol{x}$ is not known during inference, so instead, diffusion models approximate it with a neural network $\hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, \boldsymbol{c}, t) \approx \boldsymbol{x}$, where $\boldsymbol{\theta}$ is the network parameters and $\boldsymbol{c}$ denotes any given conditioning information. In practice, it is more common to indirectly parameterize $\hat{\boldsymbol{x}}_{\boldsymbol{\theta}}$ via:

$$\hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, \boldsymbol{c}, t) = \frac{1}{\alpha_t} \boldsymbol{z}_t - \frac{\sigma_t}{\alpha_t} \hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, \boldsymbol{c}, t), \tag{12}$$

which is motivated by reparameterizing Equation 9 as $\boldsymbol{z}_t = \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\varepsilon}$ for $\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{\varepsilon}; \mathbf{0}, 1)$, and then solving for $\boldsymbol{x}$. Informally, this $\boldsymbol{\varepsilon}$-parameterization learns to predict the unscaled noise that was added to a corrupted sample, while the former $\boldsymbol{x}$-parameterization learns to directly predict the denoised example itself. The marginal generative distribution under the model is then:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{c}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_0, \boldsymbol{c}) \left(\prod_{t=1}^{T} p_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t, \boldsymbol{c}) \right) p(\boldsymbol{z}_T), \tag{13}$$

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_0, \boldsymbol{c}) = \mathcal{N}\left(\boldsymbol{x}; \hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_0, \boldsymbol{c}, 0), \alpha_0^2 \sigma_0^{-2}\right), \tag{14}$$

$$p_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t, \boldsymbol{c}) = q(\boldsymbol{z}_{t-1}|\hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, \boldsymbol{c}, t), \boldsymbol{z}_t), \tag{15}$$

$$p(\boldsymbol{z}_T|\boldsymbol{c}) = \mathcal{N}(\boldsymbol{z}_T; \mathbf{0}, 1), \tag{16}$$

which can be sampled from sequentially.[2] The model is trained by minimizing a squared error loss that is simplified from a variational lower bound on the log-likelihood $\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{c})$:

$$\mathbb{E}_{\mathcal{U}(t;0,\ldots,T),\,\mathcal{N}(\boldsymbol{\varepsilon};\mathbf{0},1)}\left[\|\boldsymbol{\varepsilon}-\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\alpha_t\boldsymbol{x}+\sigma_t\boldsymbol{\varepsilon},\boldsymbol{c},t)\|_2^2\right]. \tag{17}$$

For derivations of the preceding claims, we refer readers to Ho et al. (2020).

C Derivations

C.1 Zero CoM Subspaces

Herein, we will treat point clouds $\boldsymbol{X}\in\mathbb{R}^{N\times 3}$ as vectors $\mathbf{x}=\mathrm{vec}(\boldsymbol{X})\in\mathbb{R}^{3N}$ by concatenating their columns. Assuming *normalized* point masses $\tilde{\boldsymbol{m}}\in(0,1)^N$ that sum to 1, the CoM of $\mathbf{x}$ is:

$$\boldsymbol{x}_{\text{CoM}}=(\boldsymbol{I}_3\otimes\tilde{\boldsymbol{m}}^\top)\mathbf{x}\in\mathbb{R}^3, \tag{18}$$

where $\boldsymbol{I}_3$ is the identity matrix and $\otimes$ is the Kronecker product. The set of zero CoM point clouds

$$\mathbb{U}=\{\mathbf{x}\in\mathbb{R}^{3N}\mid\boldsymbol{x}_{\text{CoM}}=\mathbf{0}\} \tag{19}$$

is an m-dimensional linear subspace, where $m=3(N-1)$, so there is an isometric isomorphism $\varphi\colon\mathbb{R}^m\to\mathbb{U}$. Using φ, we can establish a diffusion model over $\mathbb{U}$ by pushing forward a diffusion model over $\mathbb{R}^m$. In fact, due to structure-preserving properties of φ, we can train and sample over $\mathbb{U}$ without actually realizing φ or interacting with the underlying space $\mathbb{R}^m$.

Specifically, training and sampling from a standard diffusion on $\mathbb{R}^m$ (Appendix B) requires only a few types of operations in $\mathbb{R}^m$: **(1)** taking linear combinations of points, **(2)** computing the distance between two points, and **(3)** sampling from isotropic Gaussian distributions. The first two operations can equivalently be done in $\mathbb{U}$. Formally, if $\boldsymbol{u}_i\in\mathbb{R}^m$ and $\mathbf{x}_i=\varphi(\boldsymbol{u}_i)$ for $1\le i\le k$, then

$$\varphi\left(\sum_{i=1}^k\alpha_i\boldsymbol{u}_i\right)=\sum_{i=1}^k\alpha_i\mathbf{x}_i\quad\text{and}\quad\|\boldsymbol{u}_1-\boldsymbol{u}_2\|_2=\|\mathbf{x}_1-\mathbf{x}_2\|_2 \tag{20}$$

for all $\alpha_i\in\mathbb{R}$, since φ is an isomorphism and isometry, respectively. The corresponding operation for **(3)** is more involved and first requires some propositions.

Proposition 1. *Let $\boldsymbol{A}_\varphi\in\mathbb{R}^{3N\times m}$ be the matrix of φ. Then $\boldsymbol{A}_\varphi\boldsymbol{A}_\varphi^\top\in\mathbb{R}^{3N\times 3N}$ is the matrix $\boldsymbol{\Phi}$ for the orthogonal projection of $\mathbb{R}^{3N}$ onto $\mathbb{U}$.*

Proof. Since φ is an isomorphism, the columns of $\boldsymbol{A}_\varphi$ form a basis of $\mathbb{U}$. A classical result from linear algebra states that the orthogonal projection matrix can be obtained by

$$\boldsymbol{\Phi}=\boldsymbol{A}_\varphi(\boldsymbol{A}_\varphi^\top\boldsymbol{A}_\varphi)^{-1}\boldsymbol{A}_\varphi^\top=\boldsymbol{A}_\varphi\boldsymbol{A}_\varphi^\top. \tag{21}$$

The final equality follows since φ is an isometry, so $\boldsymbol{A}_\varphi^\top\boldsymbol{A}_\varphi=\boldsymbol{I}_m$ is the identity. □

Proposition 2. *Let $\boldsymbol{\Phi}\mathbf{x}$ be the orthogonal projection of $\mathbf{x}\in\mathbb{R}^{3N}$ onto $\mathbb{U}$. Then*

$$\boldsymbol{\Phi}\mathbf{x}=\mathbf{x}-\frac{1}{||\tilde{\boldsymbol{m}}||_2^2}\,\mathrm{vec}(\tilde{\boldsymbol{m}}\boldsymbol{x}_{\text{CoM}}^\top). \tag{22}$$

Proof. Evaluating $\boldsymbol{\Phi}\mathbf{x}$ is equivalent to solving the following problem over $\mathbf{p}\in\mathbb{R}^{3N}$:

$$\text{minimize: }||\mathbf{p}-\mathbf{x}||_2^2,\qquad\text{subject to: }\boldsymbol{p}_{\text{CoM}}=\mathbf{0}, \tag{23}$$

This is a quadratic minimization problem with linear equality constraints, whose solution $\mathbf{p}_\star$ satisfies

$$\begin{pmatrix}\boldsymbol{I}_{3N} & \boldsymbol{I}_3\otimes\tilde{\boldsymbol{m}}\\ \boldsymbol{I}_3\otimes\tilde{\boldsymbol{m}}^\top & \mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{p}_\star\\ \boldsymbol{\lambda}\end{pmatrix}=\begin{pmatrix}\mathbf{x}\\ \mathbf{0}\end{pmatrix} \tag{24}$$

for some Lagrange multipliers $\boldsymbol{\lambda}\in\mathbb{R}^3$. Expanding Equation 24, we have

$$\mathbf{p}_\star=\mathbf{x}-(\boldsymbol{I}_3\otimes\tilde{\boldsymbol{m}})\boldsymbol{\lambda}, \tag{25}$$

$$(\boldsymbol{I}_3\otimes\tilde{\boldsymbol{m}}^\top)\mathbf{p}_\star=\mathbf{0}. \tag{26}$$

[2] We follow Hoogeboom et al. (2022) in Equation 14.

Substituting Equation 25 into Equation 26 gives

$$\mathbf{0} = (\boldsymbol{I}_3 \otimes \tilde{\boldsymbol{m}}^\top)(\mathbf{x} - (\boldsymbol{I}_3 \otimes \tilde{\boldsymbol{m}})\boldsymbol{\lambda}) = \boldsymbol{x}_{\text{CoM}} - ||\tilde{\boldsymbol{m}}||_2^2 \boldsymbol{\lambda}, \tag{27}$$

and solving for $\boldsymbol{\lambda}$ and substituting it back into Equation 25 gives

$$\mathbf{p}_\star = \mathbf{x} - \frac{1}{||\tilde{\boldsymbol{m}}||_2^2}(\boldsymbol{I}_3 \otimes \tilde{\boldsymbol{m}})\boldsymbol{x}_{\text{CoM}} = \mathbf{x} - \frac{1}{||\tilde{\boldsymbol{m}}||_2^2}\operatorname{vec}(\tilde{\boldsymbol{m}}\boldsymbol{x}_{\text{CoM}}^\top), \tag{28}$$

as desired. □

Now, consider sampling $\boldsymbol{u} \sim \mathcal{N}(\bar{\boldsymbol{u}}, \sigma^2)$ from an isotropic Gaussian in $\mathbb{R}^m$. Let $\bar{\mathbf{x}} = \varphi(\bar{\boldsymbol{u}})$. Then the mapping of $\boldsymbol{u}$ onto $\mathbb{U}$ follows the distribution

$$\varphi(\boldsymbol{u}) \sim \mathcal{N}(\varphi(\bar{\boldsymbol{u}}), \sigma^2 \boldsymbol{A}_\varphi \boldsymbol{A}_\varphi^\top) = \mathcal{N}(\bar{\mathbf{x}}, \sigma^2 \boldsymbol{\Phi}) \tag{29}$$

by Proposition 1. Hence, we can equivalently orthogonally project a sample $\mathbf{x} \sim \mathcal{N}(\bar{\mathbf{x}}, \sigma^2)$ from an isotropic Gaussian in $\mathbb{R}^{3N}$ since

$$\boldsymbol{\Phi}\mathbf{x} \sim \mathcal{N}(\boldsymbol{\Phi}\bar{\mathbf{x}}, \sigma^2 \boldsymbol{\Phi}\boldsymbol{\Phi}^\top) = \mathcal{N}(\bar{\mathbf{x}}, \sigma^2 \boldsymbol{\Phi}). \tag{30}$$

Proposition 2 explains how to actually compute $\boldsymbol{\Phi}\boldsymbol{x}$. Indeed, in the uniform-mass case (unweighted CoM) $\tilde{\boldsymbol{m}} = \frac{1}{N}\mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^N$ is the ones-vector, Proposition 2 simplifies to

$$\boldsymbol{\Phi}\mathbf{x} = \mathbf{x} - \operatorname{vec}(\mathbf{1}\boldsymbol{x}_{\text{CoM}}^\top). \tag{31}$$

That is, $\boldsymbol{\Phi}\mathbf{x}$ is computed by subtracting the CoM from each point in $\mathbf{x}$. However, note that this is not true of the general nonuniform-mass case.

One final and minor technicality is that if our diffusion process occurs in the underlying space $\mathbb{R}^m$, our denoiser network would be a function $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\cdot, \boldsymbol{c}, t) \colon \mathbb{R}^m \to \mathbb{R}^m$. The equivalent network on the subspace $\mathbb{U}$ is a function $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}(\cdot, \boldsymbol{c}, t) \colon \mathbb{U} \to \mathbb{U}$ given by

$$\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}(\mathbf{x}, \boldsymbol{c}, t) = \varphi(\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\varphi^{-1}(\mathbf{x}), \boldsymbol{c}, t)). \tag{32}$$

However, since we want to work solely in $\mathbb{U}$, instead of directly implementing $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$, we implement $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}$ which implicitly determines $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}$ through Equation 32.

C.2 Invariance and Equivariance

Let us continue the notation and discussion from Appendix C.1, except we will now treat point clouds as $N \times 3$ matrices and also assume the conditioning features $\boldsymbol{C}$ are given as $N \times d$ matrices.

Let $\mathbb{T} \subseteq \mathrm{O}(3) = \{\boldsymbol{R} \in \mathbb{R}^{3\times 3} \mid \boldsymbol{R}^{-1} = \boldsymbol{R}^\top\}$ be a set of rigid 3D linear transformations of interest. For example, previous works (Hoogeboom et al., 2022; Xu et al., 2022) take $\mathbb{T} = \mathrm{O}(3)$ to be the space of all such motions, while we consider only the subset $\mathbb{T} = \{\operatorname{diag}(\boldsymbol{b}) \mid \boldsymbol{b} \in \{-1, +1\}^3\}$ of axially-aligned reflections. Let $\mathbb{P}$ be the set of permutation matrices $\boldsymbol{\Pi} \in \{0, 1\}^{N\times N}$ mapping $\mathbb{U}$ onto itself, so that $\boldsymbol{\Pi}\boldsymbol{X} \in \mathbb{U}$ for all $\boldsymbol{X} \in \mathbb{U}$. Then we will call distributions

$$p(\cdot \mid \cdot) \colon \mathbb{R}^m \times \mathbb{R}^{N\times d} \to \mathbb{R} \quad \text{and} \quad p(\cdot \mid \cdot, \cdot) \colon \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^{N\times d} \to \mathbb{R} \tag{33}$$

$(\mathbb{T}, \mathbb{P})$-invariant and $(\mathbb{T}, \mathbb{P})$-equivariant, respectively, if for each $\boldsymbol{R} \in \mathbb{T}$ and $\boldsymbol{\Pi} \in \mathbb{P}$, the following hold for all $\boldsymbol{u}, \boldsymbol{u}' \in \mathbb{R}^m$ and $\boldsymbol{C} \in \mathbb{R}^{N\times d}$:

$$p(\lambda(\boldsymbol{u})|\boldsymbol{\Pi}\boldsymbol{C}) = p(\boldsymbol{u}|\boldsymbol{C}) \quad \text{and} \quad p(\lambda(\boldsymbol{u})|\lambda(\boldsymbol{u}'), \boldsymbol{\Pi}\boldsymbol{C}) = p(\boldsymbol{u}|\boldsymbol{u}', \boldsymbol{C}), \tag{34}$$

where $\lambda(\boldsymbol{u}) = \varphi^{-1}(\boldsymbol{\Pi}\varphi(\boldsymbol{u})\boldsymbol{T}^\top)$. The preceding statement is saying that the likelihood of a point clouds in $\mathbb{U}$ should be unchanged regardless of how it is transformed or reordered under $(\mathbb{T}, \mathbb{P})$, but is formalized as a condition on the underlying space $\mathbb{R}^m$. In the following propositions, we prove that our diffusion model learns a $(\mathbb{T}, \mathbb{P})$-invariant generative distribution if its denoiser network satisfies a corresponding equivariance condition.

Proposition 3. *Any such λ defined above is an isometric isomorphism, so its matrix $\boldsymbol{A}_\lambda \in \mathbb{R}^{m\times m}$ is orthogonal with $|\det \boldsymbol{A}_\lambda| = 1$.*

Proof. Linearity follows from the linearity of φ and φ^{-1}. Since both are also isometries,

$$||\lambda(\boldsymbol{u})||_2^2 = \|\boldsymbol{\Pi}\varphi(\boldsymbol{u})\boldsymbol{R}^\top\|_F^2 = \|\varphi(\boldsymbol{u})\boldsymbol{R}^\top\|_F^2 = ||\varphi(\boldsymbol{u})||_F^2 = ||\boldsymbol{u}||_2^2, \tag{35}$$

where $||\cdot||_F$ is the Frobenius norm. The second equality follows since permuting the entries of a matrix does not change its norm. The third follows since applying a rigid motion to each row of a matrix does not change each row's norm and hence does not change the overall matrix norm. □

Proposition 4. *The standard Gaussian distribution $p(\boldsymbol{z}_T|\boldsymbol{C}) = \mathcal{N}(\boldsymbol{z}_T; \boldsymbol{0}, 1)$ is $(\mathbb{T}, \mathbb{P})$-invariant.*

Proof. This follows from Proposition 3 and the fact that a Gaussian distribution centered at $\boldsymbol{0}$ depends on $\boldsymbol{z}_T$ only through its norm. □

Proposition 5. *If a diffusion model has $(\mathbb{T}, \mathbb{P})$-equivariant transitions $p_{\boldsymbol{\theta}}(\boldsymbol{z}_t|\boldsymbol{z}_{t+1}, \boldsymbol{C})$ and $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_0)$, then its marginal generative distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{C})$ will be $(\mathbb{T}, \mathbb{P})$-invariant.*

Proof. Let any λ be fixed as above. Then

$$p_{\boldsymbol{\theta}}(\lambda(\boldsymbol{z}_{T-1})|\boldsymbol{\Pi}\boldsymbol{C}) = \int_{\mathbb{R}^m} p_{\boldsymbol{\theta}}(\lambda(\boldsymbol{z}_{T-1})|\boldsymbol{z}_T, \boldsymbol{\Pi}\boldsymbol{C}) p(\boldsymbol{z}_T|\boldsymbol{\Pi}\boldsymbol{C}) \mathrm{d}\boldsymbol{z}_T. \tag{36}$$

Recall that the transitions are $(\mathbb{T}, \mathbb{P})$-equivariant and the marginal distribution of $\boldsymbol{z}_T$ (which is chosen to be the standard Gaussian) is $(\mathbb{T}, \mathbb{P})$-invariant by Proposition 4. Hence,

$$p_{\boldsymbol{\theta}}(\lambda(\boldsymbol{z}_{T-1})|\boldsymbol{\Pi}\boldsymbol{C}) = \int_{\mathbb{R}^m} p_{\boldsymbol{\theta}}(\boldsymbol{z}_{T-1}|\lambda^{-1}(\boldsymbol{z}_T), \boldsymbol{C}) p(\lambda^{-1}(\boldsymbol{z}_T)|\boldsymbol{C}) \mathrm{d}\boldsymbol{z}_T. \tag{37}$$

Finally, a change of variables $\boldsymbol{u} = \lambda^{-1}(\boldsymbol{z}_T)$ gives

$$\begin{aligned}
p_{\boldsymbol{\theta}}(\lambda(\boldsymbol{z}_{T-1})|\boldsymbol{\Pi}\boldsymbol{C}) &= \int_{\mathbb{R}^m} p_{\boldsymbol{\theta}}(\boldsymbol{z}_{T-1}|\boldsymbol{u}, \boldsymbol{C}) p(\boldsymbol{u}|\boldsymbol{C}) |\det \boldsymbol{A}_\lambda| \mathrm{d}\boldsymbol{u} && (38)\\
&= \int_{\mathbb{R}^m} p_{\boldsymbol{\theta}}(\boldsymbol{z}_{T-1}|\boldsymbol{u}, \boldsymbol{C}) p(\boldsymbol{u}|\boldsymbol{C}) \mathrm{d}\boldsymbol{u} \qquad \text{(Proposition 3)} && (39)\\
&= p_{\boldsymbol{\theta}}(\boldsymbol{z}_{T-1}|\boldsymbol{C}), && (40)
\end{aligned}$$

so the marginal distribution of $\boldsymbol{z}_{T-1}$ is $(\mathbb{T}, \mathbb{P})$-invariant. We can repeat this argument inductively to find that the marginal distributions of $\boldsymbol{z}_{T-2}, \ldots, \boldsymbol{z}_0$, and $\boldsymbol{x}$, are $(\mathbb{T}, \mathbb{P})$-invariant. □

Proposition 6. *Suppose the denoiser network satisfies, for each $\boldsymbol{R} \in \mathbb{T}$ and $\boldsymbol{\Pi} \in \mathbb{P}$,*

$$\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}(\boldsymbol{\Pi}\boldsymbol{X}\boldsymbol{R}^\top, \boldsymbol{\Pi}\boldsymbol{C}, t) = \boldsymbol{\Pi}\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}(\boldsymbol{X}, \boldsymbol{C}, t)\boldsymbol{R}^\top \tag{41}$$

for all $\boldsymbol{X} \in \mathbb{U}$ and $\boldsymbol{C} \in \mathbb{R}^{N \times d}$. Then the transitions of the diffusion model are $(\mathbb{T}, \mathbb{P})$-equivariant.

Proof. By Equation 32, the denoiser on the underlying space is $\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\boldsymbol{u}, \boldsymbol{C}, t) = \varphi^{-1}(\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}^{\varphi}(\varphi(\boldsymbol{u}), \boldsymbol{C}, t))$. Direct computation shows that Equation 41 is equivalent to requiring:

$$\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\lambda(\boldsymbol{u}), \boldsymbol{\Pi}\boldsymbol{C}, t) = \lambda(\hat{\boldsymbol{\varepsilon}}_{\boldsymbol{\theta}}(\boldsymbol{u}, \boldsymbol{C}, t)). \tag{42}$$

Recall Proposition 3. It follows that:

$$\begin{aligned}
p_{\boldsymbol{\theta}}(\lambda(\boldsymbol{x})|\lambda(\boldsymbol{z}_0), \boldsymbol{c}) &= \mathcal{N}\big(\lambda(\boldsymbol{x}); \hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\lambda(\boldsymbol{z}_0), \boldsymbol{c}, 0), \alpha_0^2\sigma_0^{-2}\big) && (43)\\
&= \mathcal{N}\big(\lambda(\boldsymbol{x}); \lambda(\hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_0, \boldsymbol{c}, 0)), \alpha_0^2\sigma_0^{-2}\big) && (44)\\
&= \mathcal{N}\big(\boldsymbol{x}; \hat{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{z}_0, \boldsymbol{c}, 0), \alpha_0^2\sigma_0^{-2}\big) && (45)\\
&= p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_0, \boldsymbol{c}), && (46)
\end{aligned}$$

so the transition from $\boldsymbol{z}_0$ to $\boldsymbol{x}$ is $(\mathbb{T}, \mathbb{P})$-equivariant. The second equality follows from Equation 42 and linearity of λ. The third equality follows as a normal distribution $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \sigma^2)$ is $(\mathbb{T}, \mathbb{P})$-equivariant since it depends only on the distance $\|\boldsymbol{x} - \boldsymbol{\mu}\|_2$ and λ is an isometry. The proof of the equivariance of $p_{\boldsymbol{\theta}}(\boldsymbol{z}_t|\boldsymbol{z}_{t+1}, \boldsymbol{C})$ follows through similar manipulations. □

D Baseline Genetic Algorithm

Inspired by Mayer et al. (2019), our genetic algorithm (GA) searches over heavy atom frameworks that are consistent with a given set of unsigned substitution coordinates. Hydrogens are added later. For an example with k specified heavy atom unsigned substitution coordinates and u unavailable heavy atom substitution coordinates, each individual in the genetic population comprises a vector of binary signs $\boldsymbol{b} \in \{+, -\}^k$ and continuous positions $\boldsymbol{u} \in \mathbb{R}^u$ which, together, fully characterize a candidate heavy atom framework.

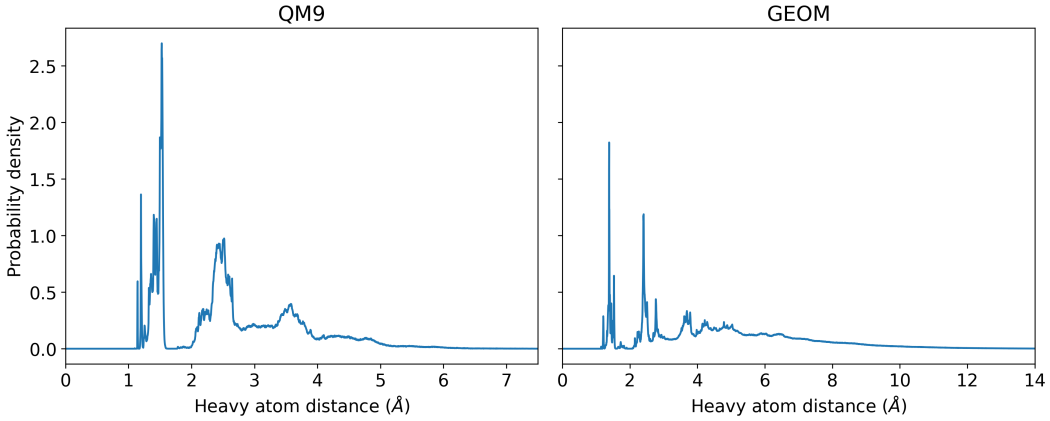

Figure 5: Pairwise heavy atom distance histograms used for the GA fitness function.

Our GA comprises a fitness, mutation, crossover, and selection function. The fitness function assigns to each candidate framework a scalar "badness" score by taking a sum of three terms:

1. **Moment error.** The Euclidean distance between the upper-triangular parts of the planar dyadic of the candidate (Equation 8) and the ground truth $\mathrm{diag}(P_X, P_Y, P_Z)$.
2. **CoM error.** The Euclidean distance of the candidate's CoM from the origin.
3. **Pairwise distance NLL.** A probability density function $\tilde{p}(d)$ of pairwise Euclidean distances between heavy atoms is estimated as a histogram from training set statistics (Figure 5). We consider the negative log-likelihood $\mathrm{NLL}(\mathbb{D}) = -\sum_{d\in\mathbb{D}} \log \tilde{p}(d)$, where $\mathbb{D}$ is the multiset of pairwise distances of atoms in a candidate framework, assuming they are independent and identically distributed (i.i.d.). Candidates with a lower NLL more closely match the training set and are more likely to make up actual molecules.

The mutation operator takes an individual and randomly bit-flips one or more elements in $\boldsymbol{b}$ and adds Gaussian noise to $\boldsymbol{u}$. The crossover operator takes in two individuals and randomly swaps selected contiguous swaths of signs between each $\boldsymbol{b}$ and does not modify $\boldsymbol{u}$. Lastly, the selection operator uses a tournament selection. Given these components, our GA minimizes the fitness function using the `eaSimple` algorithm from the Python library DEAP (Fortin et al., 2012) with 20 generations, a population size of 20,000, a mutation rate of 0.7, and a crossover rate of 0.9. At the end of the GA, the top K scoring heavy atom frameworks with unique signs $\boldsymbol{b}$ are kept.

Next, each framework is decorated with hydrogens using Hydride (Kunzmann et al., 2022), which predicts hydrogen positions by comparing heavy atom fragments to a library of hydrogen-containing fragments. Since Hydride may not necessarily add the correct number of hydrogens consistent with the molecular formula, hydrogens are randomly dropped if there are too many, and hydrogens are added with positions as Gaussian noise if there are too few. To mitigate the effects of this randomness, the adding or dropping procedure is repeated 1000 times. These hydrogen-decorated structures are scored using another function that sums two terms: **(1)** the distance of the CoM to the origin, and **(2)** $1000 \min(0, 1.09 - d_{\text{min}})^2$, where d_{min} is the minimum pairwise distance between any two atoms in the structure. This second term provides a large penalty if atoms are too close together. Only the highest scoring hydrogen-decorated structure for each heavy atom framework is kept. In total, K all-atom 3D structures are returned. On QM9 and GEOM, the GA and decoration take approximately 2 minutes per example.

E Model and Training Details

E.1 Pseudocode

Algorithm 1 Computing the training objective on a single example.

Require: The conditioning features $\boldsymbol{C} \in \mathbb{R}^{N \times d}$ and 3D conformation $\boldsymbol{X} \in \mathbb{R}^{N \times 3}$ of a molecule; a neural network $\hat{\boldsymbol{\varepsilon}}_\theta$.

Orient $\boldsymbol{X}$ to its principal axis system using Algorithm 3
$t \sim \mathcal{U}(t; 0, \ldots, T)$
$\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{\varepsilon}; \mathbf{0}, 1)$ with dimensions $\boldsymbol{\varepsilon} \in \mathbb{R}^{N \times 3}$
Orthogonally project $\boldsymbol{\varepsilon}$ onto the zero CoM subspace $\mathbb{U}$
$\boldsymbol{Z}_t \leftarrow \alpha_t \boldsymbol{X} + \sigma_t \boldsymbol{\varepsilon}$
return $\|\boldsymbol{\varepsilon} - \hat{\boldsymbol{\varepsilon}}_\theta(\boldsymbol{Z}_t, \boldsymbol{C}, t)\|_2^2$

Algorithm 2 Conditionally sampling a single example.

Require: The conditioning features $\boldsymbol{C} \in \mathbb{R}^{N \times d}$ of an unknown molecule; a neural network $\hat{\boldsymbol{\varepsilon}}_\theta$.

$\boldsymbol{Z}_T \sim \mathcal{N}(\boldsymbol{\varepsilon}; \mathbf{0}, 1)$ with dimensions $\boldsymbol{Z}_T \in \mathbb{R}^{N \times 3}$
Orthogonally project $\boldsymbol{Z}_T$ onto the zero CoM subspace $\mathbb{U}$
for $t = T$ to 1 **do**
    $\boldsymbol{Z}_{t-1} \sim p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t-1} | \boldsymbol{Z}_t, \boldsymbol{C})$
    Orthogonally project $\boldsymbol{Z}_t$ onto $\mathbb{U}$
end for
return $\boldsymbol{X} \sim p_{\boldsymbol{\theta}}(\boldsymbol{X} | \boldsymbol{Z}_0, \boldsymbol{C})$

Algorithm 3 Preprocessing a single example.

Require: The atomic masses $\boldsymbol{m} \in (0, \infty)^N$ and 3D conformation $\boldsymbol{X} \in \mathbb{R}^{N \times 3}$ of a molecule.

$M \leftarrow \sum_{i=1}^N m_i$
$\boldsymbol{x}_{\text{CoM}} \leftarrow M^{-1} \sum_{i=1}^N m_i \boldsymbol{x}_i$, where $\boldsymbol{x}_i$ is the i-th row of $\boldsymbol{X}$
$\boldsymbol{X} \leftarrow (\boldsymbol{x}_1 - \boldsymbol{x}_{\text{CoM}}, \ldots, \boldsymbol{x}_N - \boldsymbol{x}_{\text{CoM}})^\top$
$\mathbf{P} \leftarrow \sum_{i=1}^N m_i \boldsymbol{x}_i \boldsymbol{x}_i^\top$
Eigendecompose $\mathbf{P} = \boldsymbol{V} \operatorname{diag}(P_X, P_Y, P_Z) \boldsymbol{V}^\top$, where $P_X > P_Y > P_Z$
$\boldsymbol{X} \leftarrow \boldsymbol{X}\boldsymbol{V}$
$\boldsymbol{S} \leftarrow N \times 3$ binary dropout mask based on atom types and dropout rate
$|\boldsymbol{X}| \leftarrow \boldsymbol{S} \odot \text{abs}(\boldsymbol{X})$
return (
    aligned coordinates $\boldsymbol{X}$,
    partial unsigned substitution coordinates $|\boldsymbol{X}|$ with mask $\boldsymbol{S}$,
    planar moments of inertia (P_X, P_Y, P_Z)
)

Algorithm 4 Network architecture of $\hat{\boldsymbol{\varepsilon}}^{\varphi}_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{C}, t)$. We overload the notation for layers (e.g., every use of `Lin` denotes a new linear layer). See Appendix E.2 for further details.

Require: The conditioning features $\boldsymbol{C} \in \mathbb{R}^{N \times d}$ and noised 3D conformation $\boldsymbol{X} \in \mathbb{U} \subseteq \mathbb{R}^{N \times 3}$ of a molecule (centered to zero CoM); a timestep $t \in \mathbb{N}$; network parameters $\boldsymbol{\theta}$.

$\sigma \leftarrow \texttt{SiLU}$
$\boldsymbol{t} \leftarrow \texttt{SinPosEmb}(t)$ ▷ sinusoidal timestep embedding (128-dim)
$\boldsymbol{A} \leftarrow \texttt{Embed}(\boldsymbol{a})$ ▷ learned atomic number embedding (32-dim)
$\boldsymbol{C} \leftarrow (\, \boldsymbol{C} \mid \boldsymbol{A} \mid \tilde{\boldsymbol{m}} \mid \boldsymbol{t} \,)$
$\boldsymbol{H} \leftarrow \texttt{Lin}(\boldsymbol{C})$ ▷ linearly project to hidden states (256-dim)
$\boldsymbol{H}_{\text{res}} \leftarrow \boldsymbol{H}$
$\boldsymbol{C} \leftarrow \sigma(\texttt{Lin}(\sigma(\texttt{Lin}(\boldsymbol{C}))))$ ▷ MLP (dims $d_{\text{in}} \to 128 \to 128$)
$\boldsymbol{X}^{(0)} \leftarrow \boldsymbol{X}$
for i from 1 to $n_{\text{blocks}} - 1$ **do** ▷ ResiDual-like blocks ($n_{\text{blocks}} = 6$)
  $\boldsymbol{X}', \boldsymbol{H}' \leftarrow \texttt{EQBlock}(\boldsymbol{X}, \boldsymbol{H}, \boldsymbol{X}^{(0)})$
  $\boldsymbol{X} \leftarrow \boldsymbol{X}'$ with its CoM subtracted
  $\boldsymbol{H}_{\text{res}} \leftarrow \boldsymbol{H}_{\text{res}} + \boldsymbol{H}'$
  $\boldsymbol{H} \leftarrow \texttt{AdaLN}(\boldsymbol{H} + \boldsymbol{H}', \boldsymbol{C})$
  $\boldsymbol{H}' \leftarrow \texttt{Lin}(\sigma(\texttt{Lin}(\boldsymbol{H})))$ ▷ MLP (dims $256 \to 320 \to 256$)
  $\boldsymbol{H}_{\text{res}} \leftarrow \boldsymbol{H}_{\text{res}} + \boldsymbol{H}'$
  $\boldsymbol{H} \leftarrow \texttt{AdaLN}(\boldsymbol{H} + \boldsymbol{H}', \boldsymbol{C})$
  if $i = n_{\text{blocks}} - 1$ **then**
    $\boldsymbol{H} \leftarrow \boldsymbol{H} + \texttt{AdaLN}(\boldsymbol{H}_{\text{res}}, \boldsymbol{C})$
  end if
end for
$\boldsymbol{X}', \cdot \leftarrow \texttt{EQBlock}(\boldsymbol{X}, \boldsymbol{H}, \boldsymbol{X}^{(0)})$ ▷ final coordinate update
$\boldsymbol{X} \leftarrow \boldsymbol{X}'$ with its CoM subtracted
return $\boldsymbol{X} - \boldsymbol{X}^{(0)}$

E.2 Network Architecture

To satisfy Proposition 6, we implement the denoiser $\hat{\boldsymbol{\varepsilon}}^{\varphi}_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{C}, t)$ with an architecture inspired by Transformers (Vaswani et al., 2017) and E(n)-equivariant graph neural networks (EGNNs) (Satorras et al., 2021). Algorithm 4 summarizes the network architecture. The core body of the network is a ResiDual Transformer backbone (Xie et al., 2023), except the self-attention layers are replaced with an equivariant block (`EQBlock`) that jointly updates the hidden features and coordinates, which we will describe shortly. We also use a conditional version of LayerNorm (`AdaLN`) (Dieleman et al., 2022; Dhariwal and Nichol, 2021), where the affine scale and shift are given from a linear projection of some input node features $\boldsymbol{C}$, instead of being learned constants as in regular LayerNorm.

`EQBlock` is adapted from the EGNN block (Satorras et al., 2021), with the major difference being that we add reflection- but not E(3)-invariant edge features. Specifically, for $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^3$, we consider the "distance" features:

$$\Delta(\boldsymbol{x}, \boldsymbol{x}') = \big(\|\boldsymbol{x} - \boldsymbol{x}'\|_2^2, \boldsymbol{x} \cdot \boldsymbol{x}', \|\boldsymbol{x}\|_2^2, \|\boldsymbol{x}'\|_2^2, s(\boldsymbol{x} - \boldsymbol{x}'), s(\boldsymbol{x}), s(\boldsymbol{x}')\big) \in \mathbb{R}^{13}, \tag{47}$$

where $s(\boldsymbol{x}) = (x_1^2, x_2^2, x_3^2)$ denotes an element-wise squaring operation. The first four features above are E(3)-invariant, while the last three are only reflection-invariant. Features that are not translation-invariant, such as $\boldsymbol{x} \cdot \boldsymbol{x}'$, were used as suggested by Vignac et al. (2023). Then, we perform the following operations within $\texttt{EQBlock}(\boldsymbol{X}, \boldsymbol{H}, \boldsymbol{X}^{(0)})$ to produce an updated point cloud $\boldsymbol{X}'$:

$$\boldsymbol{m}_{j \to i} \leftarrow \texttt{MLP}(\boldsymbol{h}_i, \boldsymbol{h}_j, \Delta(\boldsymbol{x}_i, \boldsymbol{x}_j), \Delta(\boldsymbol{x}_i^{(0)}, \boldsymbol{x}_j^{(0)})), \quad \text{for all } i, j, \tag{48}$$

$$\boldsymbol{x}_i' \leftarrow \sum_{j=1,\, j \neq i}^{N} \left(\frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2 + 1} \right) \odot \texttt{Lin}(\boldsymbol{m}_{j \to i}), \quad \text{for all } i, \tag{49}$$

where $\boldsymbol{x}_i$, $\boldsymbol{h}$, $(\boldsymbol{x}_i^{(0)}$, and $\boldsymbol{x}_i'$ denote the i-th row of $\boldsymbol{X}$, $\boldsymbol{H}$, $\boldsymbol{X}^{(0)}$, and $\boldsymbol{X}'$; and $\odot$ denotes a Hadamard product. In the Equation 48 MLP, we use two linear layers with dimensions $d_{\text{in}} \to 320 \to 320$ and SiLU activations for each one. In Equation 49, we use a linear layer that down-projects with

dimensions $320 \to 3$. We also output node features $\boldsymbol{H}'$ using an attention-like update:

$$a_{j\to i}, \boldsymbol{v}_{j\to i} \leftarrow \texttt{Lin}(\boldsymbol{m}_{j\to i}), \quad \text{for all } i, j, \tag{50}$$

$$\boldsymbol{o}_i \leftarrow \sum_{j=1,\, j\neq i}^{N} \left(\frac{\exp(a_{j\to i})}{\sum_{k=1,\, k\neq i}^{N} \exp(a_{k\to i})} \right) \boldsymbol{v}_{j\to i}, \quad \text{for all } i, \tag{51}$$

$$\boldsymbol{h}'_i \leftarrow \texttt{Lin}(\boldsymbol{o}_i), \quad \text{for all } i, \tag{52}$$

where $\boldsymbol{h}'_i$ denotes the i-th row of $\boldsymbol{H}'$. In practice, we extend the above equations to employ multiple smaller heads (8 heads, 32-dim each), analogous to multi-headed attention.

E.3 Training and Inference Details

We use the same diffusion noise schedule and number of diffusion steps ($T = 1000$) as Hoogeboom et al. (2022). Table 2 gives the training hyperparameters of KREED on each dataset. To handle the larger molecules in GEOM, we conduct distributed training over 8 GPUs. On both datasets, we use the Adam optimizer (Kingma and Ba, 2015) with no weight decay; a linear learning rate warmup over 2000 steps (Ma and Yarats, 2021); and an adaptive gradient clipping strategy from Hoogeboom et al. (2022), whereby we clip the gradient norm at $1.5\mu + 2\sigma$, where μ and σ are the mean and standard deviation of the gradient norms of the 50 previous optimizer steps. For sampling, we use an exponential moving average (EMA) of the network parameters.

Hyperparameter	**QM9**	**GEOM**
Number of GPUs	1	8
GPU model (NVIDIA)	Quadro RTX 6000	TITAN V
GPU memory	24 GB	8×12 GB
Training time	130 h	126 h
Training steps	1.24M	0.46M
Effective batch size	512	240
Coordinate dropout $[p_{\text{min}}, p_{\text{max}}]$	$[0, 1]$	$[0, 0.5]$
Optimizer	Adam	Adam
Learning rate	4×10^{-4}	2×10^{-4}
Learning rate warmup steps	2000	2000
Gradient clipping	Yes	Yes
EMA decay	0.999	0.9995

Table 2: Training hyperparameters of KREED on QM9 and GEOM.

A detailed workflow of predicting 3D structure at inference time is depicted in Figure 6. Generating 33×100 samples for the predictions of the literature dataset required 14 minutes, or about 25.5 seconds per example, on a single NVIDIA RTX A6000 GPU with 48 GB RAM.

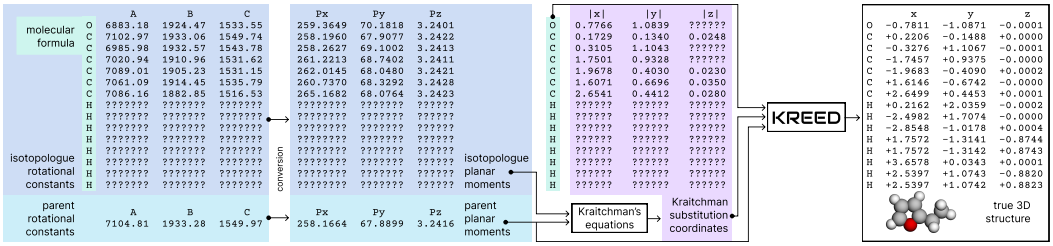

Figure 6: Explicit workflow for all-atom 3D structure prediction given molecular formula and parent and isotopologue rotational constants.

F Additional Results on QM9 and GEOM

As seen in Table 3, the genetic algorithm (GA) performs considerably better when only considering the correctness of the heavy atoms, indicating that the method of adding hydrogens is not satisfactory. However, the GA is still unable to deal with examples with many atoms or that have a significant number of missing substitution coordinates. This can be attributed to inefficient search of continuous coordinates, as well as the i.i.d. assumption of the pairwise distance NLL term breaking down for examples with many atoms.

		Correctness (%)			**Heavy Correctness (%)**		
Method	**Task**	$k = 1$	$k = 5$	$k = 10$	$k = 1$	$k = 5$	$k = 10$
Genetic algorithm	QM9	7.33	12.4	15.0	25.4	39.6	46.4
	QM9-C	0.127	0.225	0.337	0.262	0.584	0.914
	GEOM	0.0308	0.0377	0.0377	0.158	0.253	0.318
	GEOM-C	0.00342	0.00685	0.00685	0.00342	0.00685	0.0103
KREED	QM9	99.9	99.9	99.9	99.9	100.	100.
	QM9-C	91.3	93.1	93.8	92.6	94.8	95.8
	GEOM	98.9	99.2	99.2	99.4	99.5	99.5
	GEOM-C	32.6	35.8	37.0	37.0	41.6	42.9

Table 3: Top-k all-atom and heavy-atom connectivity correctness for both the GA and KREED.

F.1 No Substitution Coordinates

Even if provided with *only* molecular formula and moments, KREED can still obtain reasonable accuracy on QM9. $K = 100$ samples were generated for each example, and generated predictions were ranked by deviation from the true moments. Given that 10 times more samples were generated per example, these tasks were only evaluated for a small portion of each test set to minimize computational cost. Top-1 all-atom connectivity correctness was 27.9% for QM9 ($n = 1335$), but is 0.273% for GEOM ($n = 1464$). It was initially unexpected that accuracy on QM9 could be much greater than 0, given that this task provides very few input constraints. However, since molecules in QM9 are much smaller, it makes sense that there are only a few stable molecules with a given molecular formula and set of moments of inertia. In addition, we set $p_{\text{max}} = 1$ when training on QM9, so that some examples seen during training had almost all substitution coordinates dropped, but $p_{\text{max}} = 0.5$ when training on GEOM. It is conceivable that accuracy on this task could be raised for GEOM by increasing p_{max}, however, we suffered training instabilities when doing so.

	Correctness (%)				**Heavy Correctness (%)**			
Task	$k = 1$	$k = 5$	$k = 10$	$k = 100$	$k = 1$	$k = 5$	$k = 10$	$k = 100$
QM9	27.9	39.4	42.1	48.8	29.2	42.3	46.4	56.1
GEOM	0.273	0.342	0.342	0.546	0.273	0.410	0.478	0.820

Table 4: Top-k all-atom and heavy-atom connectivity correctness of KREED when provided with *no* substitution coordinates.

F.2 Visualizations

In the following figures, a green background indicates all-atom connectivity correctness, a yellow background indicates heavy atom connectivity correctness, and a red background indicates incorrectness. The presence of black pins indicates whether a substitution coordinate in that direction was available.

G Experimentally Measured Substitution Coordinates

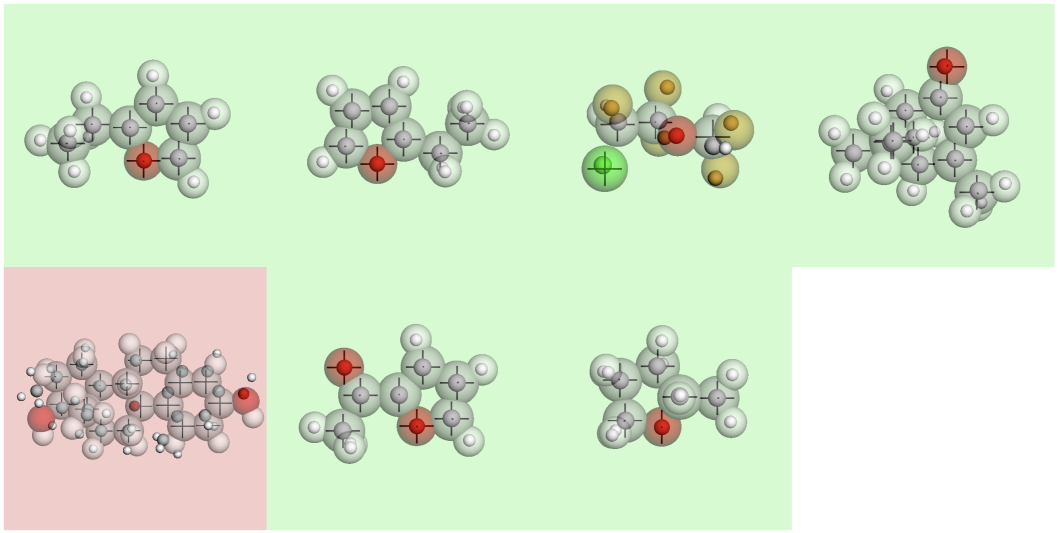

Figure 9: Top-1 predictions of KREED given experimental substitution coordinates of conformers which are very similar to examples in QM9 or GEOM.

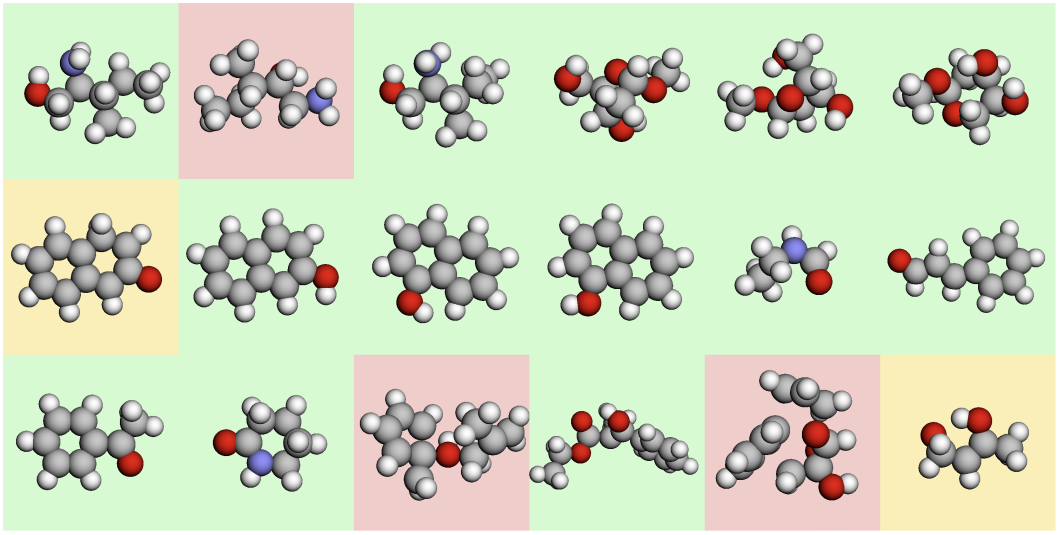

Figure 10: Top-1 predictions of KREED given experimental substitution coordinates of conformers which did not report a ground truth structure in Cartesian coordinates in their original paper. Predictions were manually verified by visual comparison to pictures in the original paper.

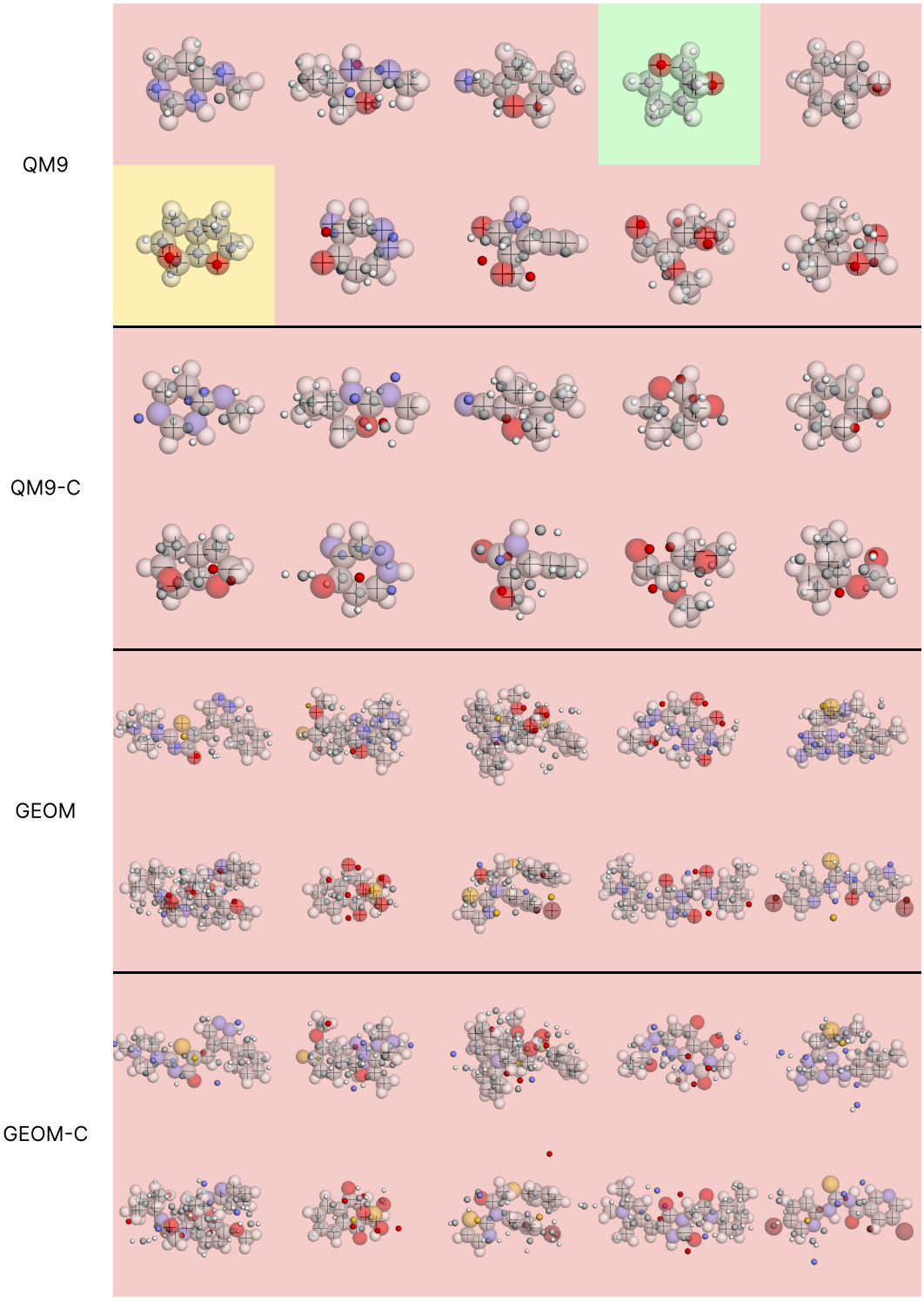

Figure 7: Top-1 predictions of the GA on random test set examples.

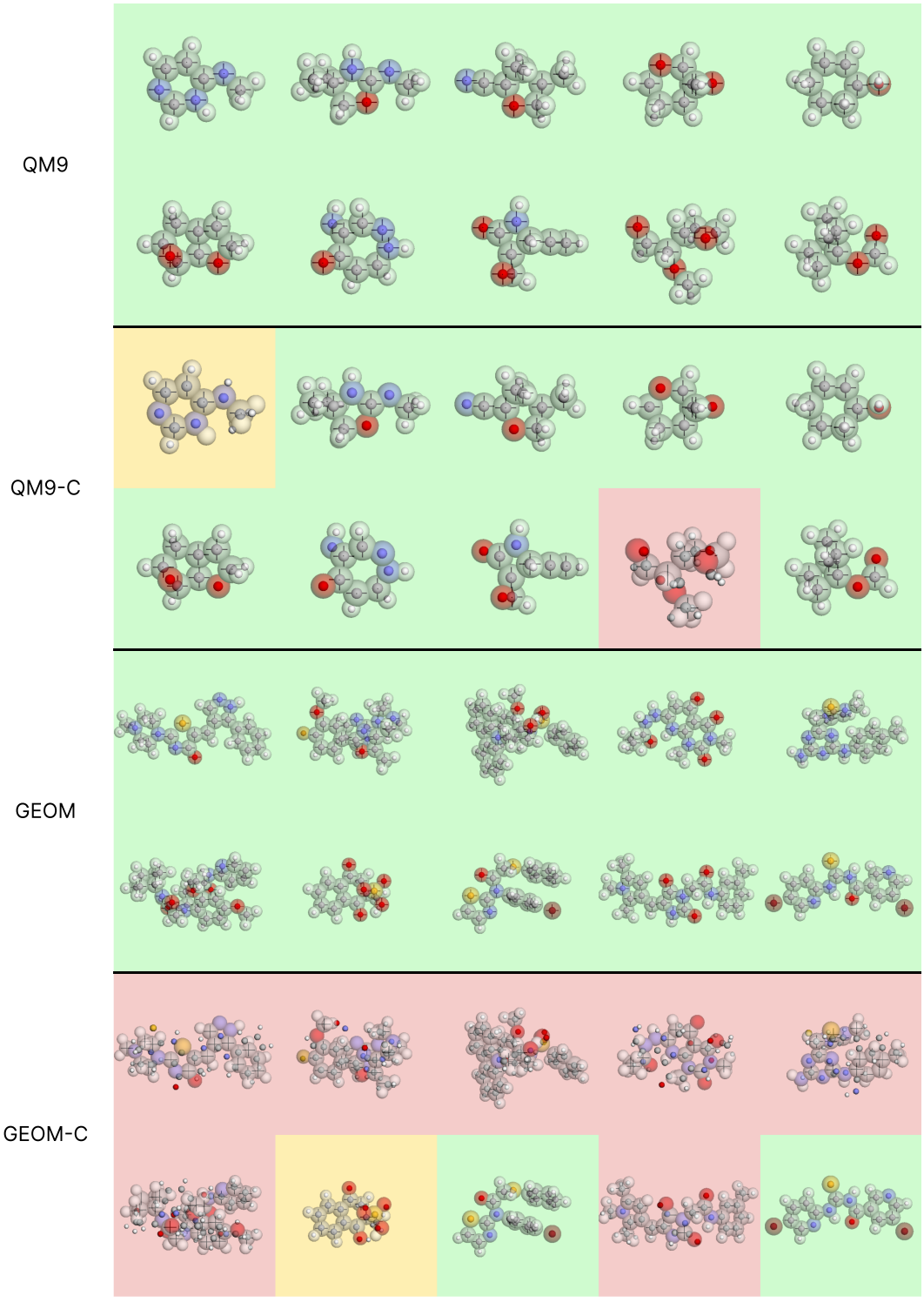

Figure 8: Top-1 predictions of KREED on random test set examples.

Conformer Name	Reference	Figure
2,3,4-trifluorotoluene	Nair et al. (2020)	Figure 4
2,4,5-trifluorotoluene	Nair et al. (2020)	Figure 4
2'-aminoacetophenone	Salvitti et al. (2022)	Figure 4
cyclopropyl (isocyanato) silane (gauche)	Guirgis et al. (2015)	Figure 4
cyclododecanone	Burevschi and Sanz (2021)	Figure 4
cycloundecanone	Tsoi et al. (2022)	Figure 4
linalool	Quesada-Moreno et al. (2019)	Figure 4
a-pinene oxide	Neeman et al. (2023)	Figure 4
2-ethylfuran (C1)	Nguyen (2020)	Figure 9
2-ethylfuran (Cs)	Nguyen (2020)	Figure 9
enflurane (G+)	Pérez et al. (2016)	Figure 9
verbenone	Marshall et al. (2017)	Figure 9
beta-estradiol (tg(+))	Zinn and Schnell (2018)	Figure 9
2-acetylfuran (anti)	Dindic et al. (2021)	Figure 9
2-methyltetrahydrofuran (equatorial)	Van et al. (2016)	Figure 9
isoleucinol (I)	Fatima et al. (2020)	Figure 10
isoleucinol (II)	Fatima et al. (2020)	Figure 10
isoleucinol (III)	Fatima et al. (2020)	Figure 10
2-deoxy-d-riboside (af-1)	Calabrese et al. (2020)	Figure 10
2-deoxy-d-riboside (bf-1)	Calabrese et al. (2020)	Figure 10
2-deoxy-d-riboside (ap-1)	Calabrese et al. (2020)	Figure 10
trans-2-naphthol	Hazrah et al. (2022)	Figure 10
cis-2-naphthol	Hazrah et al. (2022)	Figure 10
cis-1-naphthol	Hazrah et al. (2022)	Figure 10
trans-1-naphthol	Hazrah et al. (2022)	Figure 10
N-ethylformamide (trans-ac)	Ohba et al. (2005)	Figure 10
trans-cinnamaldehyde (s-trans-trans)	Zinn et al. (2015)	Figure 10
acetophenone	Lei et al. (2019)	Figure 10
delta-valerolactam	Bird et al. (2012)	Figure 10
trans-thymol-A	Quesada-Moreno et al. (2019)	Figure 10
strawberry aldehyde (t-aa)	Shipman et al. (2011)	Figure 10
strawberry aldehyde (c-sg-)	Shipman et al. (2011)	Figure 10
4-hydroxy-2-butanone (I)	Li et al. (2022)	Figure 10

Table 5: All 33 conformer examples with substitution coordinates extracted from the literature.