
Towards Joint Sequence-Structure Generation of Nucleic Acid and Protein Complexes with SE(3)-Discrete Diffusion

Alex Morehead*, Jeffrey A. Ruffolo, Aadyot Bhatnagar, & Ali Madani

AI Research Lab

Profluent Bio

Berkeley, CA, USA

acmwhb@missouri.edu, {jruffolo, abhatnagar, ali}@profluent.bio

Abstract

Generative models of macromolecules carry abundant and impactful implications for industrial and biomedical efforts in protein engineering. However, existing methods are currently limited to modeling protein structures or sequences, independently or jointly, without regard to the interactions that commonly occur between proteins and other macromolecules. In this work, we introduce MMDIFF, a generative model that jointly designs sequences and structures of nucleic acid and protein complexes, independently or in complex, using joint SE(3)-discrete diffusion noise. Such a model has important implications for emerging areas of macromolecular design including structure-based transcription factor design and design of noncoding RNA sequences. We demonstrate the utility of MMDIFF through a rigorous new design benchmark for macromolecular complex generation that we introduce in this work. Our results demonstrate that MMDIFF is able to successfully generate micro-RNA and single-stranded DNA molecules while being modestly capable of joint modeling DNA and RNA molecules in interaction with multi-chain protein complexes. Source code: <https://github.com/Profluent-Internships/MMDiff>.

1 Introduction

The ability to effectively design proteins using computational methods is a long-sought-after goal of protein engineering research [22]. Such techniques could enable accelerated development of vaccines targeting novel viruses, enhanced exploration of the space of designable materials, and the ability to engineer new protein-based energy resources towards a more sustainable climate future. Recent progress in computationally designing protein variants given an initial wildtype sequence has demonstrated the utility of machine learning in protein engineering [10]. More recently, deep learning models based on structure prediction networks have demonstrated the ability to flexibly design novel protein structures with desired functions in wet-lab experiments [25]. Nonetheless, considerably less attention has been directed toward the design of novel protein-nucleic acid interactions, which carry implications for computationally engineering new means of transcription factors and intervening in a host of essential functions within living organisms. Approaching such design tasks is arguably more challenging, as the number of high-quality macromolecular complexes currently available in the RCSB Protein Data Bank (PDB) [4] is notably scarce while the shapes of such complexes can also be much more flexible compared to proteins.

Towards this end, in this work we explore the design of new nucleic acid (NA) and protein complexes using a generative diffusion process we refer to as SE(3)-discrete diffusion. In Section 4.2, we

* Alex Morehead was employed by Profluent Bio as a research intern at the time of this work.

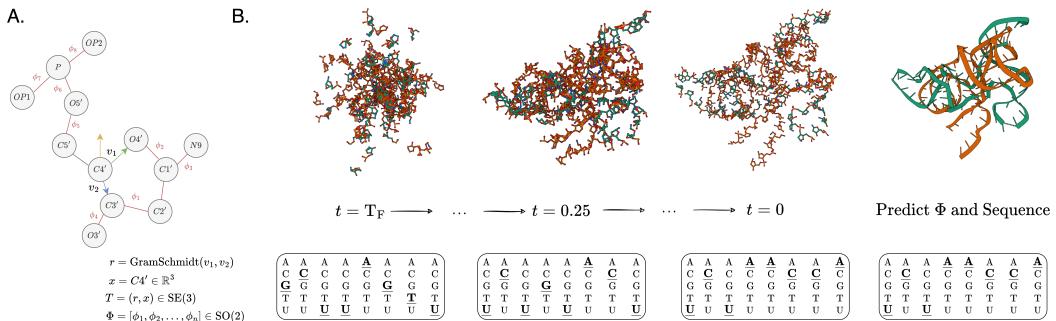


Figure 1: Our proposed Macromolecular Diffusion Model (MMDIFF) jointly designs macromolecular sequences and structures. **A.** An overview of MMDIFF. For each nucleic acid residue, a rigid body frame centered at the $C4'$ atom is constructed. To build such a frame, the GramSchmidt algorithm is applied to a residue’s v_1 and v_2 vectors, in the process placing its $C3'$, $O4'$, and $C5'$ atoms with respect to the position of the $C4'$ atom. The positions of all other residue atoms are placed autoregressively according to a corresponding torsion angle (Φ) predicted by MMDIFF. **B.** An illustrative example of how MMDIFF generates realistic macromolecular samples. Through the iterative process of denoising N geometric frames and N one-hot sequence vectors initialized from their respective reference distributions, MMDIFF transitions an initially-random sequence-structure pair at timestep T_F into a coherent macromolecule at timestep 0, at which point each frame and its associated torsion angles are used to construct the position of each atom.

discuss the $SE(3)$ -discrete process which guides deep learning models to jointly reverse an $SE(3)$ diffusion process² on rigid body frames in \mathbb{R}^3 [28] and a related diffusion process on discrete sequence inputs [5]. In Section 4.3, we introduce our proposed $SE(3)$ -discrete diffusion method for protein-nucleic acid design, MMDIFF, which is to our best knowledge the first generative diffusion model for the design of macromolecular sequences and structures beyond proteins. Through our experiments in Section 2, we introduce a systematic open-source suite of computational benchmarks to assess the ability of a generative macromolecular design model to produce designable, diverse, and novel samples, and, under such a framework, highlight the utility of MMDIFF for macromolecular complex generation. Consequently, our contributions presented in this work pave the way for the development of future macromolecular generative models.

2 Experiments

Metrics. To investigate the performance of MMDIFF (and similar generative models developed in the future) in generating new macromolecular complexes, we propose to examine the quality of its generated complexes from multiple perspectives: (1) the model’s ability to generate *designable* sequences such that, upon predicting the structure of such a designed sequence using an external structure prediction model, the difference between the model’s co-designed structure and the sequence-predicted structure is minimal; (2) the model’s ability to generate *diverse* complex structures that are distinct from each other after structure-based clustering; and (3) the model’s ability to generate (3) *novel* complex structures that are well-differentiated from complex structures found within the model’s training dataset.

Designability. To instantiate such metrics in the context of macromolecular complex generation as outlined above, for (1) we adopt RoseTTAFold2NA [3] for sequence-based prediction of generated protein-nucleic acid structures to assess *designability*, where we report structural similarities in terms of self-consistency RMSD and TM-score (i.e., scRMSD and scTM) for each complex as determined by US-align [29]. Due to practical concerns regarding the significant computational requirements of running RoseTTAFold2NA for each complex, in this setting, we generate and evaluate one sequence per structure produced by the model, using RoseTTAFold2NA in *single-sequence* mode to reduce its average prediction time from 25 minutes to 2 minutes. Note that this may limit the sequence

² $SE(3)$ refers to the group of 3D rotations and translations (excluding reflections), whereas $SE(3)^N$ represents the manifold of N $SE(3)$ frames.

Method	$< 5\text{\AA}$ scRMSD	D_{chain}	$D_{complex}$	D_{single}	D_{all}	N_{pool}
Random Generation	0.00%	0.92	1.00	1.00	1.00	0.85
MMDIFF	0.74%	0.92	1.00	1.00	1.00	0.72
MMDIFF-PROTEIN	0.00%	0.91	0.99	0.99	0.99	0.73

(a) Protein Generation

Method	$< 5\text{\AA}$ scRMSD	D_{chain}	$D_{complex}$	D_{single}	D_{all}	N_{pool}
Random Generation	1.33%	0.99	1.00	0.99	0.99	0.75
MMDIFF	8.67%	1.00	1.00	1.00	1.00	0.74
MMDIFF-NA	6.00%	0.98	0.98	0.99	0.99	0.65

(b) Nucleic Acid Generation

Method	$< 5\text{\AA}$ scRMSD	D_{chain}	$D_{complex}$	D_{single}	D_{all}	N_{pool}
Random Generation	0.37%	1.00	1.00	1.00	1.00	0.72
MMDIFF	0.74%	0.96	0.85	1.00	1.00	0.69
MMDIFF-MONOMER	0.00%	0.96	0.85	1.00	1.00	0.70

(c) Protein-Nucleic Acid Complex Generation

Table 1: A comparison of our proposed method, MMDIFF, against a random baseline for generating different types of biological complexes, where we assess each method’s designability (based on an scRMSD threshold), diversity (regarding four variants of D), and novelty (i.e., N_{pool}). Across all tasks, MMDIFF represents the same model trained on protein-nucleic acid complexes. However, in task (a), MMDIFF-PROTEIN is trained on both protein monomers and protein complexes; in task (b), MMDIFF-NA is trained solely on nucleic acid complexes; and lastly, in task (c), MMDIFF-MONOMER is trained on both monomers and protein-nucleic acid complexes.

designability performance that MMDIFF can achieve, as previous works have found it necessary to design up to 8 sequences per generated structure to achieve good designability results. Moreover, RoseTTAFold2NA was trained to predict protein-nucleic acid structures using multiple sequence alignments (MSAs) as a crucial input feature, hence running the model in *single-sequence* mode will incur some prediction accuracy degradation in exchange for practical running times. Taking all of these constraints into account, we propose to evaluate each method’s ability to generate molecules with an scRMSD $< 5 \text{ \AA}$ to denote a successfully-designed macromolecule, with an scRMSD $< 2 \text{ \AA}$ being a more ideal threshold [28] yet being largely impractical due to the potential mismatches between generated and predicted inter-molecule interaction types (e.g., RF2NA predicting non-interacting chain structures and MMDIFF generating interacting chain structures).

Diversity. Regarding (2), we use qTMclust [29] for macromolecular structure-based clustering of the model’s pool of generated complex structures to measure *diversity*, where the diversity of a pool is defined in four components as follows. We define the baseline chain diversity (D_{chain}) of a pool as: (number of chain clusters) / (number of chains). Similarly, as an upper bound on the model’s diversity, we define the simpler complex diversity ($D_{complex}$) of a pool as: (number of complex clusters) / (number of complexes). More specifically, to measure how often the model generates a unique interaction type, we then define the single – chain complex diversity (D_{single}) as: (number of complexes for which one of its chains is a chain cluster representative) / (number of complexes). Lastly, as a lower bound on the model’s diversity, we define the all – chain complex diversity (D_{all}) as: (number of complexes for which each of its chains is a chain cluster representative) / (number of complexes).

Novelty. To address (3) we perform pairwise structural alignments between the model’s pool of generated complex structures and the model’s training dataset using US-align [29] to estimate *novelty*, where the novelty of a pool is quantified as the inverse of the maximum TM-score between a generated complex structure and any complex structure in the training dataset (i.e., $N_{pool} = 1 - \text{maxTM}$).

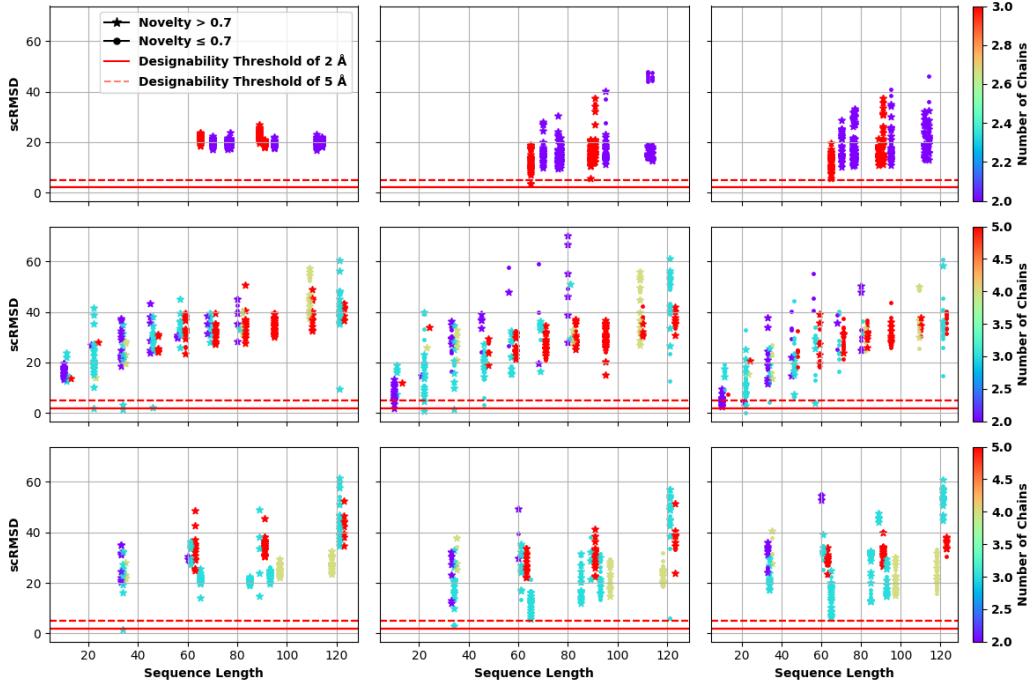
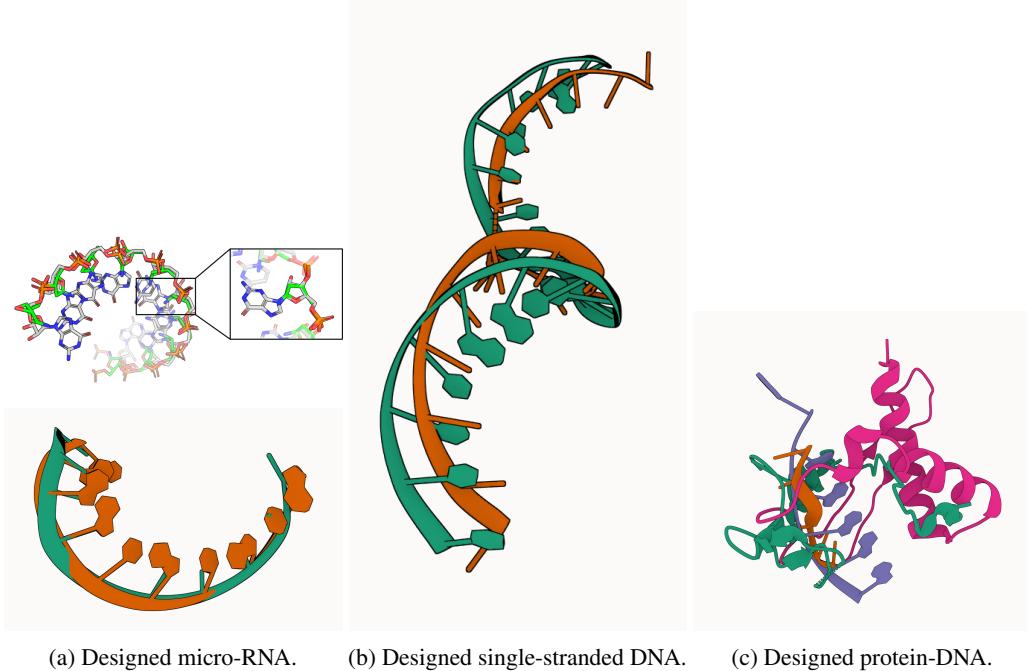


Figure 2: Comparison of scRMSD complex designability results using different training methods. Here, the top row corresponds to protein-only experiments, the middle row to nucleic acid-only experiments, and the bottom row to protein-nucleic acid experiments. The columns denotes samples generated using the random macromolecule generation baseline, MMDIFF, and MMDIFF- {PROTEIN, NA, MONOMER } (corresponding to the {first, second, third} row), respectively. Note that novel data samples are displayed with a * symbol. Overall, most designable complexes contain 2-3 chains, and most generated complexes contain novel chains (with a novelty > 0.7).

Baselines. To the best of our knowledge, no prior methods can co-design sequences and structures of macromolecular complexes. As such, in addition to evaluating MMDIFF, we report results using a naive baseline that randomly generates sequences and structures of protein, nucleic acid, and protein-nucleic acid complexes, respectively, as a simple baseline to evaluate a macromolecular generative model’s performance. For the random baseline, we extract the initial sequence and structure noise representations of MMDIFF’s diffusion process (i.e., noise representations at timestep T) as the random baseline’s generated outputs. As such, one should expect a generative model that has learned meaningful sequence and structural patterns from its training dataset to outperform such a baseline regarding the designability of its samples by fully reversing its learned diffusion process.

Results and Analysis. As shown in Table 1a and Figure 2, the results for protein complex generation suggest that MMDIFF (trained on protein-nucleic acid complexes) can successfully generate a handful of designable protein complexes, whereas the random baseline cannot. Interestingly, MMDIFF-PROTEIN, a version of MMDIFF that is trained not only on protein-nucleic acid complexes but also jointly on the large monomeric protein structure dataset of [28], underperforms MMDIFF in all design metrics. One explanation for this observation is that the generative modeling space that MMDIFF-PROTEIN is tasked to model when jointly training on protein monomers is much larger and noisier compared to the modeling space of solely protein-nucleic acid complexes, a notion that is supported by the striking dataset distribution shift observed between Figures 7 and 10.

Table 1b shows that both MMDIFF (trained on protein-nucleic acid complexes) and MMDIFF-NA (trained solely on nucleic acid complexes) outperform the random baseline method for nucleic acid complex generation in terms of designability (i.e., scRMSD and scTM). Important to note is that MMDIFF generalizes markedly beyond the designability of the random baseline by generating a reasonable number of moderately designable, highly diverse, and novel complex structures. Figure 2 illustrates the designability scores MMDIFF can achieve in terms of scRMSD across a



(a) Designed micro-RNA. (b) Designed single-stranded DNA. (c) Designed protein-DNA.

Figure 3: Examples of macromolecules successfully designed by MMDIFF.

variety of complex lengths and chain compositions. Lastly, Figures 3a and 3b illustrate micro-RNA (GCGCGCGGGG) and single-stranded micro-DNA (TCTTGTTGTCTTCGA) molecules successfully designed using MMDIFF, which achieve scRMSD measures of 1.26 and 4.95 and maximum trainTM measures of 0.36 and 0.31 with respect to the PDB complexes 4WTF and 1XPX, respectively. Moreover, RoseTTAFold2NA is confident in its predicted structure for these designed sequences, yielding per-residue average pLDDT scores of 95.11 and 78.24, respectively. Note, for illustrating these molecules, the corresponding MMDIFF-generated structures are shown in green (a) or orange (b), whereas the RF2NA-predicted structures are shown in orange (a) or green (b).

Table 1c and Figure 2 show that MMDIFF outperforms the random baseline as well as MMDIFF-MONOMER, similar to the results listed in Table 1a for protein generation. We argue this result is also likely due to increased modeling space complexity introduced by adding a majority of protein monomers to the training dataset of MMDIFF-MONOMER. Figure 3c illustrates a protein-DNA complex (DNA sequence: GGGGGG) designed by MMDIFF. Note that, here, MMDIFF generates a structurally-novel protein-DNA interaction in comparison to that of RF2NA while maintaining the main structure of the DNA molecule (scRMSD of DNA structure $< 5 \text{ \AA}$), where RF2NA's per-residue average pLDDT of 58.10 may indicate the possibility of multiple conformations for such an interaction type (as suggested by MMDIFF). Note, for visualizing this macromolecule, the MMDIFF-generated chains are shown in green (orange), whereas the RF2NA-predicted chains are shown in pink (purple). Also, note that all corresponding designability results in terms of scTM are displayed in Appendix 4.5 within Figure 4.

3 Discussions & Conclusions

In this work, we introduced MMDIFF, a generative model for nucleic acid and protein complex sequences and structures, which sets the stage for the development of future macromolecular generative models. Preliminary results with MMDIFF demonstrate its ability to generate diverse and novel nucleic acid structures in computational benchmarking. However, it faces limitations in achieving success with protein complex and joint protein-nucleic acid design, suggesting a need for improved training data. As such, future work could involve (1) curating larger high-quality macromolecular datasets; (2) investigating new diffusion noise schedules; (3) validating nucleic acid designs experimentally; or (4) exploring efficient scoring methods for complex molecular assemblies.

Acknowledgments and Disclosure of Funding

The authors would like to thank Jason Yim for insightful discussions regarding the SE(3) Diffusion framework and its successful applications in protein backbone design. This work was funded by Profluent Bio and performed while Alex Morehead was a summer research intern with the company.

References

- [1] N. Anand and T. Achim. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. *arXiv preprint arXiv:2205.15019*, 2022.
- [2] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. van den Berg. Structured denoising diffusion models in discrete state-spaces. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 17981–17993. Curran Associates, Inc., 2021.
- [3] M. Baek, R. McHugh, I. Anishchenko, D. Baker, and F. DiMaio. Accurate prediction of nucleic acid and protein-nucleic acid complexes using rosettafoldna. *bioRxiv*, pages 2022–09, 2022.
- [4] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.
- [5] T. Chen, R. Zhang, and G. Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022.
- [6] R. R. Eguchi, C. A. Choe, and P.-S. Huang. Ig-vae: Generative modeling of protein structure by direct 3d coordinate generation. *PLoS computational biology*, 18(6):e1010271, 2022.
- [7] R. Evans, M. O'Neill, A. Pritzel, N. Antropova, A. Senior, T. Green, A. Žídek, R. Bates, S. Blackwell, J. Yim, et al. Protein complex prediction with alphafold-multimer. *biorxiv*, pages 2021–10, 2021.
- [8] C. Fu, K. Yan, L. Wang, W. Y. Au, M. McThrow, T. Komikado, K. Maruhashi, K. Uchino, X. Qian, and S. Ji. A latent diffusion model for protein structure generation. *arXiv preprint arXiv:2305.04120*, 2023.
- [9] A. Gelbin, B. Schneider, L. Clowney, S.-H. Hsieh, W. K. Olson, and H. M. Berman. Geometric parameters in nucleic acids: sugar and phosphate constituents. *Journal of the American Chemical Society*, 118(3):519–529, 1996.
- [10] B. L. Hie and K. K. Yang. Adaptive machine learning for protein engineering. *Current Opinion in Structural Biology*, 72:145–152, 2022.
- [11] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [12] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34:12454–12465, 2021.
- [13] E. Hoogeboom, V. G. Satorras, C. Vignac, and M. Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR, 2022.
- [14] C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer, and A. Rives. Learning inverse folding from millions of predicted structures. In *International Conference on Machine Learning*, pages 8946–8970. PMLR, 2022.
- [15] A. Jamasb, R. Viñas Torné, E. Ma, Y. Du, C. Harris, K. Huang, D. Hall, P. Lió, and T. Blundell. Graphein-a python library for geometric deep learning and network analysis on biomolecular structures and interaction networks. *Advances in Neural Information Processing Systems*, 35:27153–27167, 2022.

- [16] C. K. Joshi, A. R. Jamasb, R. Viñas, C. Harris, S. Mathis, and P. Liò. Multi-state rna design with geometric multi-graph neural networks. *arXiv preprint arXiv:2305.14749*, 2023.
- [17] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.
- [18] X. Li, J. Thickstun, I. Gulrajani, P. S. Liang, and T. B. Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022.
- [19] Y. Lin and M. AlQuraishi. Generating novel, designable, and diverse protein structures by equivariantly diffusing oriented residue clouds. *arXiv preprint arXiv:2301.12485*, 2023.
- [20] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, A. dos Santos Costa, M. Fazel-Zarandi, T. Serbu, S. Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv*, 2022:500902, 2022.
- [21] S. L. Lisanza, J. M. Gershon, S. W. K. Tipps, L. Arnoldt, S. Hendel, J. N. Sims, X. Li, and D. Baker. Joint generation of protein sequence and structure with rosettafold sequence space diffusion. *bioRxiv*, pages 2023–05, 2023.
- [22] S. Lutz. Beyond directed evolution—semi-rational protein engineering and design. *Current Opinion in Biotechnology*, 21(6):734–743, 2010. Chemical biotechnology – Pharmaceutical biotechnology.
- [23] T. Shen, Z. Hu, Z. Peng, J. Chen, P. Xiong, L. Hong, L. Zheng, Y. Wang, I. King, S. Wang, et al. E2efold-3d: end-to-end deep learning method for accurate de novo rna 3d structure prediction. *arXiv preprint arXiv:2207.01586*, 2022.
- [24] B. L. Trippe, J. Yim, D. Tischer, D. Baker, T. Broderick, R. Barzilay, and T. Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. *arXiv preprint arXiv:2206.04119*, 2022.
- [25] J. L. Watson, D. Juergens, N. R. Bennett, B. L. Trippe, J. Yim, H. E. Eisenach, W. Ahern, A. J. Borst, R. J. Ragotte, L. F. Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, pages 1–3, 2023.
- [26] K. E. Wu, K. K. Yang, R. van den Berg, J. Zou, A. X. Lu, and A. P. Amini. Protein structure generation via folding diffusion. *arXiv preprint arXiv:2209.15611*, 2022.
- [27] M. Xu, A. S. Powers, R. O. Dror, S. Ermon, and J. Leskovec. Geometric latent diffusion models for 3d molecule generation. In *International Conference on Machine Learning*, pages 38592–38610. PMLR, 2023.
- [28] J. Yim, B. L. Trippe, V. De Bortoli, E. Mathieu, A. Doucet, R. Barzilay, and T. Jaakkola. Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*, 2023.
- [29] C. Zhang, M. Shine, A. M. Pyle, and Y. Zhang. Us-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes. *Nature methods*, 19(9):1109–1115, 2022.

4 Supplementary Material

4.1 Preliminaries and Notation

Protein geometry parametrization. Inspired by the recent work of [28], we adopt the standard backbone parametrization of AlphaFold 2 for protein geometry modeling. Succinctly, with such a parametrization, N_p protein residues are modeled as N_p rigid body frames centered at each residue’s C_α^* atom ($C_\alpha^* = (0, 0, 0)$), with each frame representing a residue’s N^* , C_α^* , C^* , and O^* (local) atoms, respectively. For a given protein residue indexed by n , to obtain the absolute (global) positions of its main atoms, one may apply a special Euclidean ($SE(3)$) transformation T_n as

$$[N_n, C_n, (C_\alpha)_n] = T_n \cdot [N^*, C^*, C_\alpha^*], \quad (1)$$

where $T_n = (r_n, x_n)$ is composed of two components, $r_n \in SO(3)$ being a 3×3 rotation matrix and $x_n \in \mathbb{R}^3$ being a translation vector. Subsequently, all N_p frames are denoted as $\mathbf{T}_p = [T_1, \dots, T_{N_p}] \in SE(3)^{N_p}$. Lastly, one may place each residue’s O^* atom by rotating it around the corresponding residue’s $C_\alpha - C$ bond using the torsion angle ψ . For additional details regarding the mapping between idealized and absolute protein atom positions, we refer readers to [28].

Nucleic acid geometry parametrization. To model nucleic acid residues jointly with protein residues, we require an expressive frame encoding of nucleic acid geometry. Towards this end, we propose to model N_n nucleic acid residues as N_n rigid body frames centered at each residue’s $C4'^*$ atom ($C4'^* = (0, 0, 0)$), with each frame representing a residue’s $C1'^*$ through $C5'^*$ atoms, $O3'^*$ through $O5'^*$ atoms, as well as its P^* , $OP1^*$, $OP2^*$, and $N9^*$ (or, for pyrimidine residues, $N1^*$) atoms, respectively. Then for a given nucleic acid residue indexed by n , to obtain the absolute positions of its main atoms, one may apply an $SE(3)$ transformation $T_n = (r_n, x_n)$ as

$$[C3'_n, O4'_n, C4'_n, C5'_n] = T_n \cdot [C3'^*, O4'^*, C4'^*, C5'^*], \quad (2)$$

where all N_n frames are denoted as $\mathbf{T}_n = [T_{N_p+1}, \dots, T_{N_n}] \in SE(3)^{N_n}$. For nucleic acid residues, to place each residue’s auxiliary atoms (i.e., $C2'^*, C1'^*, N9^*$ (or $N1^*$), $O3'^*, O5'^*, P^*, OP1^*, OP2^*$), one may autoregressively rotate them around the corresponding residue’s $C4' - C3'$, $C4' - O4'$, $O4' - C1'$, $C4' - C3'$, $C4' - C5'$, $C5' - O5'$, $O5' - P'$, and $O5' - P'$ bonds using the torsion angles $\Phi = [\phi_1, \dots, \phi_8]$, respectively. For additional background regarding nucleic acid geometry, we refer readers to [9].

Joint protein-nucleic acid geometry parametrization. Given N_p protein frames and N_n nucleic acid frames, respectively, we subsequently denote the manifold of frames that we aim to represent with a generative model as $\mathbf{N} = (N_p, N_n)$. Consequently, we recover the original means of modeling a distribution $\mathbf{X}^{(0)} \sim p_0$ of frames in $SE(3)^N$ supported on a Riemannian manifold \mathcal{M} , as proposed by [28]. In short, using a score network $s_\theta(t, \cdot)$, this allows one to model $\mathbf{X}^{(0)}$ via an approximation of the Stein score $\nabla \log p_t$ using the $SE(3)$ denoising score matching (DSM) loss [28]:

$$\mathcal{L}(\theta) = \mathbb{E}[\lambda_t \|\nabla \log p_{t|0}(\mathbf{X}^{(t)} | \mathbf{X}^{(0)}) - s_\theta(t, \mathbf{X}^{(t)})\|^2], \quad (3)$$

where θ are the network parameters to be optimized by minimizing \mathcal{L} ; $p_{t|0}$ is the density of $\mathbf{X}^{(t)}$ given $\mathbf{X}^{(0)}$; $\lambda_t > 0$ is a loss weight; and the expectation \mathbb{E} is taken over $(\mathbf{X}^{(0)}, \mathbf{X}^{(t)})$ and $t \sim \mathcal{U}([0, T_F])$, with T_F being the final timestep in a diffusion (i.e., noising) process over $\mathbf{X}^{(0)}$.

Remaining preliminaries. Following the notation of [28], bold denotes the concatenation of variables (e.g., $\mathbf{n} = (n_1, \dots, n_N)$); uppercase denotes stochastic variables (e.g., $X \sim p$), whereas lowercase is reserved for deterministic variables; and superscripts with parentheses denote time (e.g., $x^{(t)}$).

4.2 Joint Continuous-Discrete Diffusion in \mathbb{R}^3

Sequence modeling. Expressively modeling arbitrary macromolecular structures is a non-trivial task for a generative model to address. Nonetheless, in this work, we argue that an understanding of macromolecules is inherently incomplete without also considering the sequence space of such molecules. In particular, modeling of macromolecular sequences enables methods to learn molecular representations that are complementary, and in many cases foundational, to their structure knowledge

of 3D molecular structures [17, 20, 3, 23]. This is particularly relevant for nucleic acid modeling in that, practically speaking, no general-purpose existing methods currently exist for fixed-backbone sequence design of nucleic acid structures, although RNA-specific sequence design methods are beginning to emerge [16]. As such, in this work, we are also interested in modeling the design space of macromolecular sequences. We do so by introducing a discrete diffusion framework that, with key adaptations to its noise scheduling, is amenable to the SE(3) Diffusion framework of [28] described above in Section 4.1. This allows our proposed method, MMDIFF, to jointly generate sequence-structure pairs of new macromolecules, thereby circumventing the need for fixed-backbone sequence design methods for nucleic acid residues given a generated backbone structure.

Discrete sequence generation. Diffusion generative modeling of discrete data types such as amino acid and nucleic acid sequences has received considerably less attention in the research literature compared to diffusion generative approaches to modeling continuous data such as images or even 3D point cloud coordinates. Nonetheless, preliminary works have investigated alternative approaches to expressively modeling discrete values in a diffusion generative setting [12, 2, 5], with some directly adding discrete noise to one’s discrete network inputs and others first representing discrete data in a continuous space and then applying standard Gaussian diffusion techniques to generate new discrete data samples through the use of an ARGMAX operation [11]. In this work, we adopt the latter approach of modeling discrete macromolecular sequence data as continuous representations such that (1) methodological developments for continuous diffusion models will be directly amenable to such an approach and (2) it then becomes trivial to condition sample generation on arbitrary sequence features of interest (e.g., secondary structure annotations for protein residues) by embedding such features in the same continuous space as the input sequence. Concretely, inspired by [21], we propose to model a discrete amino acid and nucleic acid sequence \mathbf{S}_d in a shared vocabulary space (i.e., $\mathbb{R}^{s \times n}$) as a zero-centered continuous one-hot vector representation

$$\mathbf{S}_c^{(0)} = 2 * \text{onehot}(\mathbf{S}_d) - 1 = \{2 * (s \in \{0.0, 1.0\}^n : \sum_{i=1}^n s_i = 1.0) - 1, \forall s \in \mathbf{S}_d\}, \quad (4)$$

such that discrete sequences can be generated by reversing a continuous Gaussian diffusion process

$$q(\mathbf{S}_c^{(t_d)} | \mathbf{S}_c^{(0_d)}) = \mathcal{N}(\mathbf{S}_c^{(t_d)} | \sqrt{\bar{\alpha}^{(t_d)}} \mathbf{S}_c^{(0_d)}, (1 - \bar{\alpha}^{(t_d)}) \mathbf{I}) \quad (5)$$

on an initially-random sequence vector representation $\mathbf{S}_c^{(T_d)}$ corresponding to the final sequence diffusion timestep T_d . Here, $\bar{\alpha}^{(t_d)}$ represents $1 - \beta^{(t_d)}$, where $\beta^{(t_d)}$ denotes one’s chosen variance schedule for the sequence diffusion process, and t_d represents a discretized version of the structure diffusion continuous timestep t via equally-spaced binning of such timesteps. Note that this binning operation is important to ensure that the sequence diffusion and structure diffusion processes are in alignment with one another at each step in their respective diffusion trajectories.

Training and sampling with structure noise. Reversal of the structure noise schedule is performed by first using the FrameDiff algorithm of [28] to predict at each structure diffusion timestep the denoised frames $\mathbf{F}^{(0)}$ to derive the respective rotation (r) and translation (x) scores $\{(\zeta_{\theta,n}^r, \zeta_{\theta,n}^x)\}_{n=1}^N$, where during model training such score predictions are supervised using Equation 3 along with supervision for predicted torsions Φ through auxiliary backbone atom and inter-atom distance losses. The scores $\{(\zeta_{\theta,n}^r, \zeta_{\theta,n}^x)\}_{n=1}^N$ are then used to transition to a previous structure diffusion timestep as $\mathbf{F}^{(t-1)} = \{\exp_{\mathbf{F}_n^{(t)}}\{(W_n^r, W_n^x)\}\}_{n=1}^N$, where (W_n^r, W_n^x) are derived independently as $W_n^x = P[P\gamma[\frac{1}{2}X_n^{(t)} + \zeta_{\theta,n}^x] + \zeta\sqrt{\gamma}[Z_n^x \sim \mathcal{N}(0, \text{Id}_3)]]$ and $W_n^r = \gamma\zeta_{\theta,n}^r + \zeta\sqrt{\gamma}[Z_n^r \sim \mathcal{T}\mathcal{N}_{R_n^{(t)}}(0, \text{Id})]$, respectively [28]. Here, $P \in \mathbb{R}^{3N \times 3N}$ represents a projection matrix that removes an object’s center of mass; $\gamma = \frac{1-\epsilon}{N_{\text{steps}}}$ with ϵ being the final continuous timestep of the structure diffusion noise schedule prior to predicting torsions Φ ; $X_n^{(t)} \in \mathbb{R}^3$ and $R_n^{(t)} \in \mathbb{R}^{3 \times 3}$ denote the translation vector and rotation matrix associated with frame (i.e., residue) $\mathbf{F}_n^{(t)}$ at continuous timestep t ; Z_n^x and Z_n^r denote Gaussian noise sampled from the tangent spaces of $X_n^{(t)}$ and $R_n^{(t)}$, respectively; and $\zeta \in [0, 1]$ represents a scaling factor for structure diffusion noise.

Training and sampling with sequence noise. As pointed out by [18, 21], in the context of training diffusion models for discrete sequence generation it is also desirable to use a square root diffusion noise schedule for $\beta^{(t_d)}$, as using other diffusion noise schedules during training may make it trivial for a model to predict a ground-truth sequence from a small-noise discrete timestep t_d using the

ARGMAX operation. Such prior works thereby sensitize the model to sequence noise at small t_d timesteps by training it with a square root noise schedule, while supervising its sequence predictions using a categorical cross-entropy loss. After model training with a square root noise schedule, in this work we also explore sampling sequences from a trained model using cosine and linear noise schedules instead, to investigate how doing so affects the model at inference time with regards to its sequence generation trajectories (and thereby its structure generation trajectories), especially at small t_d timesteps. Additional experiments in Appendix 4.6 show that using sequence noise schedules other than a square root noise schedule during sampling produces marginally more designable nucleic acid complex structures overall, which suggests an interesting avenue for future research. Reversal of the sequence noise schedule is then performed by having a trained model directly predict $\mathbf{S}_c^{(0_d)}$ at timestep t_d and then using this value in Equation 5 to transition to timestep $t_d - 1$.

4.3 Generating macromolecular complexes with MMDIFF

As illustrated in Figure 1, our proposed method, MMDIFF, enables joint generation of nucleic acid sequences and structures through a hybrid SE(3)-discrete diffusion process. Moreover, by way of its design as described in Section 4.1, MMDIFF can simultaneously model protein sequences and structures together with nucleic acids, enabling the generation of inter-molecular complexes. In the remainder of this section, we will discuss the remaining details necessary to train and sample from MMDIFF for generating macromolecular complexes.

Model architecture, data, and featurization. Inspired by its demonstrated ability to generate designable, diverse, and novel monomeric protein structures, for MMDIFF we adapt the FrameDiff model architecture of [28] for modeling of macromolecular complexes. To train MMDIFF for macromolecular modeling, we downloaded from the RCSB PDB [4] all available instances of protein-nucleic acid structures using the PDBMANAGER API within the GRAPHEIN Python package [15]. We filtered the downloaded structures to those that (1) were determined at a resolution of 4.5 Å or better using nuclear magnetic resonance, diffraction, or cryo-electron microscopy; (2) contain protein chains between lengths 40 and 256 or nucleic acid chains between lengths 3 and 256; and (3) contain no more than 10 protein or 10 nucleic acid chains. In Appendix 4.7, we plot the length and chain composition distributions of the various training dataset configurations we explore in this work. To accurately characterize the intricate chain compositions present within many macromolecular PDB complexes, we subsequently adopt the relative position encoding of AlphaFold-Multimer [7] in contrast to the absolute position encoding used by [28]. Otherwise, we adapt the feature initialization scheme of [24] to accommodate both proteins and nucleic acids.

Intra-molecule consensus sampling. To generate internally-consistent macromolecular sequences, we propose a consensus sampling algorithm that, during sampling, constrains MMDIFF to select only amino acid residue types in a designated protein sequence region and only the residues types of a single type of nucleic acid molecule (i.e., either DNA or RNA) in a nucleic acid sequence region. Namely, for a given sequence representation $\mathbf{S}_c^{(t_d)}$, we propose to consensus-sample sequences by dynamically modifying the pseudo-probabilities (i.e., continuous representation) of $\mathbf{S}_c^{(t_d)}$ such that the largest probabilities associated with residue types outside of the current sequence’s molecule type consensus (e.g., $C^{(t_d)} = [\text{DA}, \text{DC}, \text{DG}, \text{DT}] = \text{DNA residue types}$ if 50% or more of the generated residues are DNA residues) are set to be marginally smaller than the smallest probabilities associated with residue types within the current molecule type consensus as

$$\mathbf{S}_c^{(t_d)} = \{s \in \mathbb{R}^n : \max_{i \notin C^{(t_d)}} s_i < \min_{i \in C^{(t_d)}} s_i, \forall s \in \mathbf{S}_c \text{ with } i \in [1, n]\}. \quad (6)$$

Applying Equation 6 after each iteration of sequence denoising during sampling effectively biases the generative sequence distribution from which MMDIFF samples new sequences, such that the model is increasingly less likely to naturally sample out-of-vocabulary within a given chain.

4.4 Related Work

Generative methods for molecule generation. Several kinds of generative models have been proposed for molecular structure generation [13, 6, 27] and sequence generation [14], most commonly diffusion generative models [1, 26, 24, 19, 8, 25]. Regarding diffusion generative models, most closely related to this work is the SE(3) diffusion model of [28] and the joint protein sequence-structure

diffusion model of [21]. The former is designed solely to generate protein backbone structures, and the latter method uses a pre-trained structure prediction network to design protein sequence-structure pairs using pure Gaussian noise. In contrast to these methods, MMDIFF designs sequence-structure pairs of not only proteins but also nucleic acid complexes using a geometrically-expressive rigid body denoising process for structure generation and a discrete diffusion process for sequence generation, thereby deriving the benefits of both data representations for macromolecular inputs while avoiding the need for pre-training.

Nucleic acid modeling. Few works have rigorously explored geometric modeling of nucleic acid structures to the extent that methods such as AlphaFold 2 [17] have done for amino acid residues. Some preliminary works in this direction are those of [23, 3, 16]. Nonetheless, no prior works have presented a geometric modeling scheme for nucleic acids that is amenable to macromolecular design tasks while being complementary to amino acid modeling, which we present in this work.

4.5 Additional Designability Results

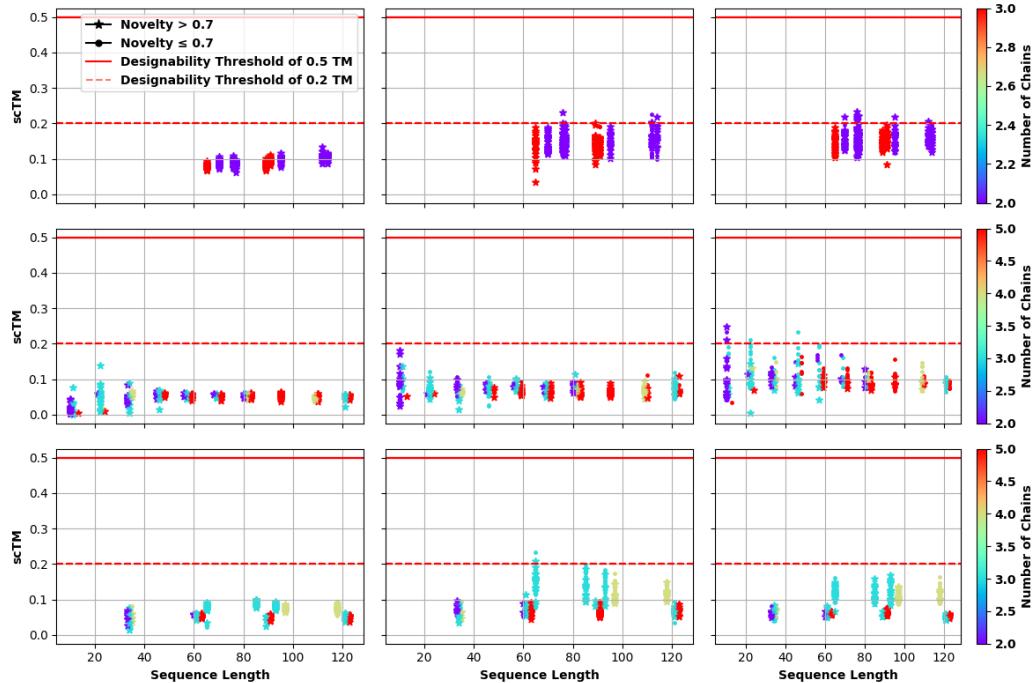
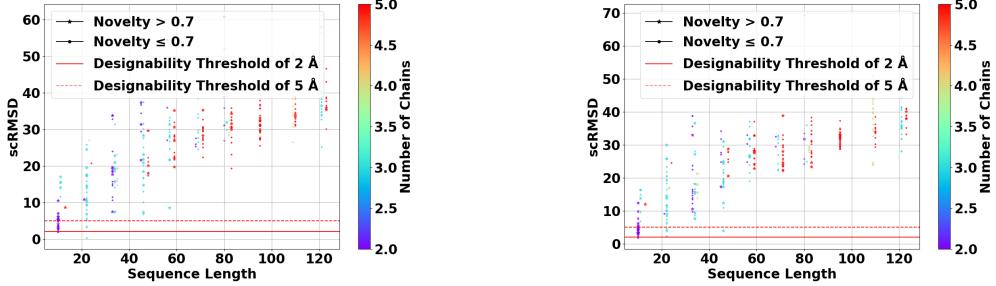


Figure 4: Comparison of scTM complex designability results using different training methods. Here, the top row corresponds to protein-only experiments, the middle row to nucleic acid-only experiments, and the bottom row to protein-nucleic acid experiments. The columns denote samples generated using the random macromolecule generation baseline, MMDIFF, and MMDIFF-{ PROTEIN, NA, MONOMER } (corresponding to the {first, second, third} row), respectively. Note that novel data samples are displayed with a * symbol. Overall, most designable complexes contain 2-3 chains, and most generated complexes contain novel chains (with a novelty > 0.7).

In Figure 4, we report designability results in terms of scTM for different training methods as a complement to the results in Figure 2.

4.6 Alternative Noise Schedules for Sequence Generation

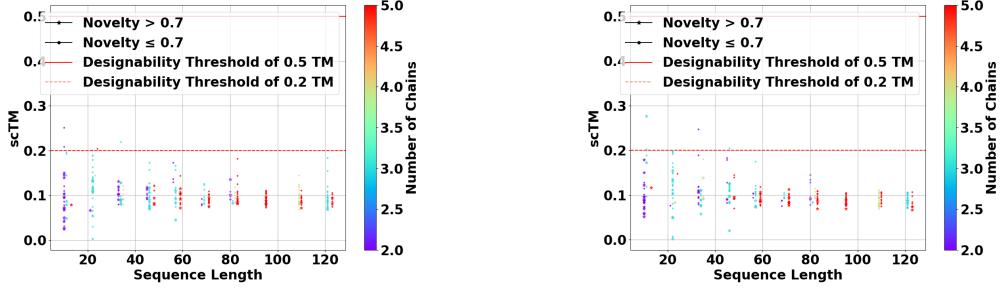
As Figures 5 and 6 illustrate, in terms of designability metrics, linear and cosine sequence noise schedules marginally outperform a square root noise schedule, producing slightly more designable macromolecules overall. However, to simplify the presentation of results, in the main text we report MMDIFF generation results using a square root noise schedule.



(a) scRMSD designability results for nucleic acid structures using a version of MMDIFF trained only on nucleic acid complexes and sampled using linear sequence noise.

(b) scRMSD designability results for nucleic acid structures using a version of MMDIFF trained only on nucleic acid complexes and sampled using cosine sequence noise.

Figure 5: Comparison of scRMSD complex designability results for nucleic acid structures using different sequence noise schedules during sampling. Zoom in for the best viewing experience.



(a) scTM designability results for nucleic acid structures using a version of MMDIFF trained only on nucleic acid complexes and sampled using linear sequence noise.

(b) scTM designability results for nucleic acid structures using a version of MMDIFF trained only on nucleic acid complexes and sampled using cosine sequence noise.

Figure 6: Comparison of scTM complex designability results for nucleic acid structures using different sequence noise schedules during sampling. Zoom in for the best viewing experience.

4.7 Dataset Distributions

Figure 7 displays the length distribution of protein-nucleic acid complexes within MMDIFF’s protein-nucleic acid training dataset, whereas Figures 8 and 9 display (in isolation) the length distributions of protein complexes and nucleic acid complexes, respectively, within the protein-nucleic acid training dataset. Likewise, Figure 10 illustrates the length distribution of protein-nucleic acid complexes (and protein monomers) within MMDIFF’s training dataset after combining it with the monomeric protein structure dataset of [28].

Lastly, to investigate how well MMDIFF can model the true radius of gyration distribution for nucleic acid structures in its training dataset, Figures 11a and 11b display the distributions of (1) MMDIFF’s training dataset and (2) MMDIFF’s generated samples. As shown in these figures, MMDIFF generates its samples with an average radius of gyration of 1.8, which is reasonably close to the 1.65 average radius of gyration of its training dataset.

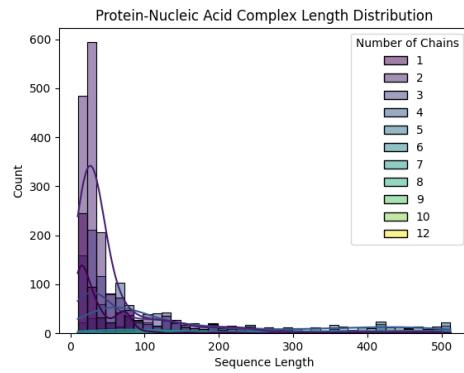


Figure 7: The distribution of protein-nucleic acid complex lengths upon which MMDIFF is trained.

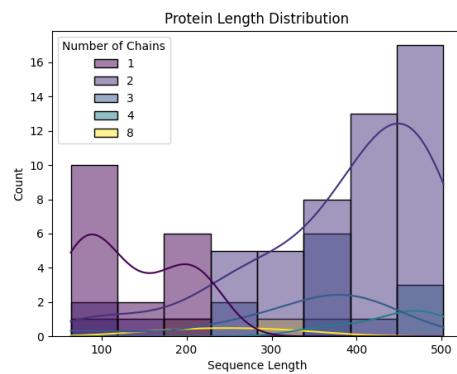


Figure 8: The distribution of protein monomer and protein complex lengths upon which MMDIFF is trained.

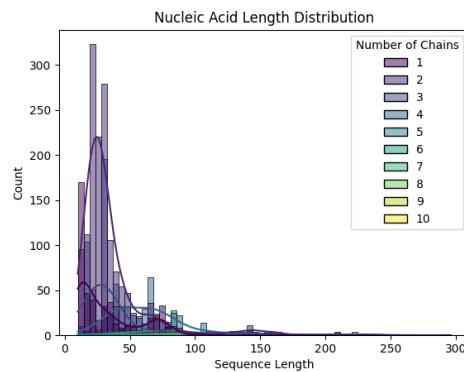


Figure 9: The distribution of nucleic acid complex lengths upon which MMDIFF is trained.

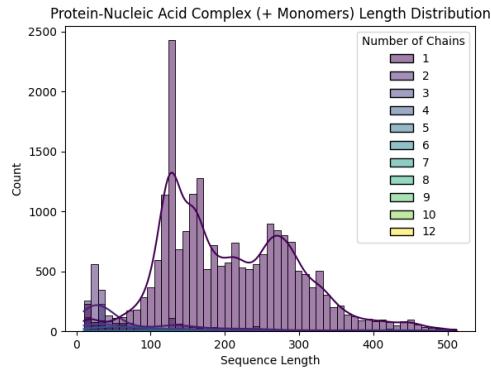
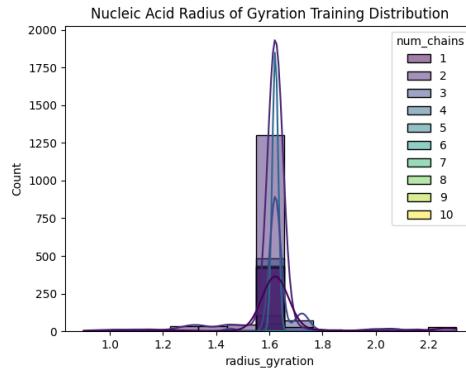
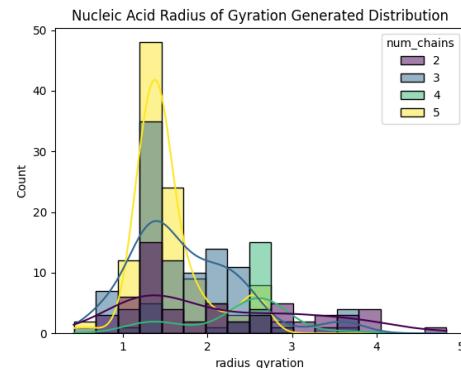


Figure 10: The distribution of protein-nucleic acid complex (and monomer) lengths upon which MMDIFF-MONOMER (for protein-nucleic acid complex generation) is trained.



(a) The radius of gyration training distribution for nucleic acid structures.



(b) The radius of gyration generated distribution for nucleic acid structures.

Figure 11: A comparison between the radius of gyration distributions of MMDIFF’s training dataset and MMDIFF’s generated samples.