
Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
Физтех-школа Прикладной Математики и Информатики
Кафедра интеллектуальных систем

Направление подготовки / специальность: 03.04.01 Прикладные математика и физика

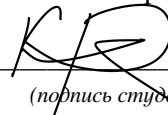
Направленность (профиль) подготовки: Математическая физика, компьютерные технологии и
математическое моделирование в экономике

ПОРОЖДАЮЩИЕ МОДЕЛИ БЕЛКОВЫХ СТРУКТУР

(магистерская диссертация)

Студент:

Клыпа Роман Сергеевич



(подпись студента)

Научный руководитель:

Грудинин Сергей Владимирович,
канд. физ.-мат. наук



(подпись научного руководителя)

Консультант (при наличии):

(подпись консультанта)

Москва 2024

Аннотация

В данной дипломной работе представлен новый генеративный процесс на специальной ортогональной группе $SO(3)$, который является важным инструментом для моделирования вращений в трехмерном пространстве. Основное внимание уделено разработке и анализу этого процесса, а также доказательству его ключевых свойств, таких как сходимость и эквивариантность.

В работе также приведены экспериментальные результаты, демонстрирующие преимущества предложенного генеративного процесса. На синтетических данных показано, что новый метод превосходит существующие подходы по точности и стабильности. Более того, проведенные эксперименты на реальной задаче условной генерации трехмерных структур молекул РНК подтверждают сходимость и практическую применимость предложенного процесса.

Таким образом, предложенный генеративный процесс в пространстве $SO(3)$ обладает значительным потенциалом для различных приложений в области молекулярной биологии и других сфер, требующих точного моделирования трехмерных структур.

Содержание

1	Введение	4
1.1	Представление трехмерных объектов	4
1.2	Генеративные процессы	4
2	Литературный обзор	4
3	Постановка задачи	6
4	Предлагаемый подход	6
5	Вычислительный эксперимент	9
6	Генерация трехмерных структур РНК	10
6.1	Представление молекул РНК и белков	10
6.2	MolBindDif	11
6.3	Encoder	12
6.4	Функция потерь	13
6.5	Обучающая, валидационная и тестовая выборки	13
6.6	Детали обучения	14
6.7	Atom Restoration Network	14
7	Анализ результатов	15
7.1	Вычислительный эксперимент	15
7.2	Генерация трехмерных структур РНК	16
8	Заключение	16
	References	18
A	Доказательства и выводы	21
A.1	Вывод формулы 26	21
A.2	Доказательство Леммы 4.4	21
A.3	Доказательство Леммы 4.6	22
B	Дополнительные графические материалы	22
B.1	Архитектура модели	22
B.2	Результат MolBindDif	24

1 Введение

Генерация трёхмерных (3D) объектов представляет собой важное направление в компьютерной графике, архитектуре, дизайне и многих других областях. В последние годы наблюдается значительный прогресс благодаря машинному обучению и нейронным сетям, которые позволяют создавать реалистичные 3D объекты на основе 2D изображений или текстовых описаний. Генерация молекул в 3D позволяет значительно ускорить процесс разработки новых лекарств и материалов, делая исследования более точными и эффективными.

1.1 Представление трехмерных объектов

Один из способов представления трехмерных объектов - это использование облаков точек (point clouds), где каждый объект представлен множеством точек в пространстве \mathbb{R}^3 . Однако наличие дополнительной информации, например, о связях между этими точками, позволяет группировать их в твёрдые тела и описывать в терминах пространства $SE(3)^N$, где N - количество отдельных твёрдых тел. Такое представление открывает новые возможности для более точного моделирования и анализа 3D объектов, включая их трансформации и взаимодействия в трёхмерном пространстве, что особенно важно для задач, связанных с генерацией сложных структур, таких как молекулы.

1.2 Генеративные процессы

Генеративные процессы играют ключевую роль в создании трёхмерных объектов, обеспечивая методы для автоматического формирования сложных структур. Эти процессы включают различные подходы, такие как вариационные автоэнкодеры (VAE) [1], генеративно-состязательные сети (GAN) [2], нормализующие потоки [3] и согласование градиентов [4]. В данной работе мы сосредотачиваемся на методе согласования градиентов. Согласование градиентов (score matching) — это метод обучения параметров модели, который минимизирует расхождение между истинной плотностью распределения данных и плотностью, задаваемой моделью, за счёт минимизации разности градиентов логарифмов этих плотностей:

$$\mathcal{L}(\theta) = \mathbb{E}_{p(\mathbf{x})} [\|s_{\theta}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p(\mathbf{x})\|^2]. \quad (1)$$

Этот метод эффективен для сложных многомерных данных, так как напрямую работает с градиентами, обеспечивая более точное соответствие между моделью и реальными данными. Имея доступ к градиентам, сэмплирование из распределения модели можно осуществлять с использованием методов динамики Ланжевена:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \epsilon \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \sqrt{2\epsilon} \mathbf{z}_i, \quad i = 0, 1, \dots, K, \quad \mathbf{z}_i \sim \mathcal{N}(0, I). \quad (2)$$

2 Литературный обзор

В отличие от евклидова пространства, на $SE(3)^N$ не существует канонического прямого процесса диффузии. Наиболее полное теоретическое обоснование для диффузионных процессов на $SE(3)$ на данный момент изложено в статье [5]. В ней авторы показывают,

что при соответствующем выборе метрики группа $SE(3)$ может быть идентифицирована с $SO(3) \times \mathbb{R}^3$ с точки зрения римановой геометрии, что позволяет определить оператор Лапласа-Бельтрами и броуновское движение. Такой выбор имеет преимущество простоты и позволяет рассматривать процессы на \mathbb{R}^3 и $SO(3)$ независимо. Для \mathbb{R}^3 авторы предлагают следующий прямой процесс:

$$d\mathbf{X}^{(t)} = f_x(t)\mathbf{X}^{(t)} + g_x(t)d\mathbf{B}_{\mathbb{R}^3}^{(t)}, \quad (3)$$

где $f_x(t)$ - коэффициент дрейфа, $f_x(t) = -\frac{1}{2}\beta(t)$ и $g_x(t)$ - коэффициент диффузии, $g_x(t) = \sqrt{\beta(t)}$ для некоторого $\beta(t)$. $\mathbf{B}_{\mathcal{M}}^{(t)}$ является Броуновским движением на многообразии \mathcal{M} .

Этот процесс можно рассмотреть как масштабированный по времени процесс Орнштейна-Уленбека [6]. Как следствие, полагая $G_x(t) = \int_0^t g_x(s)^2 ds$, мы получаем распределение условной вероятности:

$$p_{t|0}(\mathbf{X}^{(t)}|\mathbf{X}^{(0)}) = \mathcal{N}(\mathbf{X}^{(t)}; \exp^{-G_x(t)}\mathbf{X}^{(0)}, 1 - \exp^{-G_x(t)}), \quad (4)$$

где \mathcal{N} - нормальное распределение. Для распределения вращений $\mathbf{R}^{(t)}$ авторы предлагают процесс

$$d\mathbf{R}^{(t)} = g_r(t)d\mathbf{B}_{SO(3)}^{(t)}, \quad (5)$$

где коэффициент диффузии $g_r(t) = \sqrt{\frac{d}{dt}\sigma^2(t)}$ для некоторого $\sigma(t)$. Это дает нам следующее условное распределение:

$$p_{t|0}(\mathbf{R}^{(t)}|\mathbf{R}^{(0)}) = \text{IGSO}_3(\mathbf{R}^{(t)}; \mathbf{R}^{(0)}, \sigma^2(t)), \quad (6)$$

где $\text{IGSO}_3(\mathbf{R}^{(t)}; \mathbf{R}^{(0)}, \sigma^2(t)) = \text{IGSO}_3(\mathbf{R}^{(0)\top}\mathbf{R}^{(t)}, \sigma^2(t))$ - изотропное распределение Гаусса на $SO(3)$ [7]. Это распределение может быть параметризовано в ось-угловой форме, с равномерно выбранными осями и углом вращения $\omega \in [0, \pi]$ с плотностью

$$f(\omega, t) = \sum_{l \in \mathbb{N}} (2l+1) e^{-l(l+1)\sigma^2(t)/2} \frac{\sin((l+1/2)\omega)}{\sin(\omega/2)}. \quad (7)$$

Соответствующие 3 и 5 обратные процессы выглядят следующим образом:

$$d\mathbf{X}^{(t)} = (g_x(t)^2 s^x(\mathbf{X}^{(t)}, t) - f_x(t))dt + \zeta g_x(t)d\mathbf{B}_{\mathbb{R}^3}^{(t)}, \quad (8)$$

$$d\mathbf{R}^{(t)} = g_r(t)^2 s^r(\mathbf{R}^{(t)}, t)dt + \zeta g_r(t)d\mathbf{B}_{SO(3)}^{(t)}, \quad (9)$$

где $s^x(\mathbf{X}^{(t)}, t)$ и $s^r(\mathbf{R}^{(t)}, t)$ могут быть рассчитаны следующим образом:

$$s^x(\mathbf{X}^{(t)}, t)_n = \nabla_{x_n^{(t)}} \log p_{t|0}(x_n^{(t)}|x_n^{(0)}) = -\frac{x_n^{(t)} - e^{-\frac{1}{2}\beta(t)}x_n^{(0)}}{1 - e^{\beta(t)}}, \quad (10)$$

$$s^r(\mathbf{R}^{(t)}, t)_n = \nabla_{r_n^{(t)}} \log p_{t|0}(r_n^{(t)}|r_n^{(0)}) = -\frac{r_n^{(t)}}{\omega(r_n^{(0)})} \log r_n^{(0,t)} \partial_\omega f(\omega(r_n^{(0)}), t). \quad (11)$$

Здесь n - номер элемента в $SE(3)^N$, $r_n^{(t)}$ и $x_n^{(t)}$ - его соответствующие поворот и трансляция.

Однако процесс 5, предложенный авторами для $SO(3)$, страдает от неполного зашумления:

$$p_{1|0}(\mathbf{R}^{(1)}|\mathbf{R}^{(0)}) = \text{IGSO}_3(\mathbf{R}^{(1)}; \mathbf{R}^{(0)}, \sigma^2(1)). \quad (12)$$

Для генерации изображений было показано [8], что неполное зашумление может являться причиной предвзятости итоговой модели. В статье [9] было показано, что подобный процесс при достаточно больших $\sigma(t)$ сходится к $\mathcal{U}_{SO(3)}$, однако на практике такая сходимость может не наблюдаться.

3 Постановка задачи

В данной работе мы стремимся построить новые генеративные прямой и обратный процессы на $SO(3)$. Для обратного процесса нами был выбран метод согласования градиентов плотности распределения. Данные прямой $\mathbf{R}_t(\mathbf{R}_0)$ и обратный $\mathbf{R}_t(\mathbf{R}_1)$ процессы должны обладать рядом свойств:

1. Эквивариантность ($g_1 f(g_2) = f(g_1 g_2)$): прямой и обратный процессы должны быть эквиварианты по отношению к действию группы $SO(3)$.
2. Полное зашумление: прямой процесс при $t = 1$ не должен зависеть от \mathbf{R}_0 . Распределение $p_1(\mathbf{R}_1(\mathbf{R}_0))$ должно быть доступным для выборки.
3. Трассируемость распределения прямого процесса: распределение $(\mathbf{R}_t(\mathbf{R}_0))$ должно быть известным и выражаться в явном виде.
4. Сходимость: необходимы теоретические гарантии сходимости обратного процесса.

Поскольку свойства большинства потенциальных трехмерных объектов, принадлежащих $SE(3)^N$, инварианты относительно действия группы $SE(3)$ (при условии одинакового действия на каждый из N элементов), эквивариантность генеративных процессов должна упрощать обучение модели, предсказывающей градиенты плотности распределений.

4 Предлагаемый подход

Основная идея нашего подхода заключается в использовании отображений из группы вращений в соответствующую ей алгебру Ли $\log : SO(3) \rightarrow \mathfrak{so}(3)$ и обратного $\exp : \mathfrak{so}(3) \rightarrow SO(3)$. Здесь и далее мы обозначаем \mathbf{R} элементы принадлежащие $SO(3)$ и \mathbf{r} - принадлежащие $\mathfrak{so}(3)$. Концепция касательного пространства $\mathfrak{so}(3)$ к группе $SO(3)$ позволяет рассматривать малые изменения (инфинитезимальные приращения) элементов этой группы. Элементы касательного пространства представляют собой касательные векторы, которые указывают направление изменения элементов группы. Используя экспоненциальное отображение, мы можем двигаться вдоль геодезических на группе $SO(3)$ между двумя ее элементами.

Алгебра Ли для группы $SO(3)$ имеет представление в виде антисимметричных матриц 3×3 с базисом

$$\mathbf{L}_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{L}_y = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad \mathbf{L}_z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (13)$$

Любой матрице из $\mathfrak{so}(3)$ можно сопоставить вектор Эйлера $\boldsymbol{\omega} = (x, y, z)$:

$$\mathbf{r} = x\mathbf{L}_x + y\mathbf{L}_y + z\mathbf{L}_z = \mathbf{L}_z = \begin{bmatrix} 0 & -x & z \\ x & 0 & -y \\ -z & y & 0 \end{bmatrix}. \quad (14)$$

Удобство векторов Эйлера заключается в возможности изменять угол поворота, домножая вектор на соответствующее число. На практике большинство прямых процессов зашумления на \mathbb{R}^N выглядят как постепенное смещение изначального элемента к нулю с добавлением шума. Исходя из желания иметь схожий процесс для $SO(3)$, мы выдвигаем следующее предложение:

Proposition 4.1. *Прямой процесс на $SO(3)$:*

$$\mathbf{R}_t = \exp[d(t)\mathbf{r}_0 * \tilde{\mathbf{r}}(t)], \quad \tilde{\mathbf{r}}(t) \sim \text{IGSO}_3(\mathbf{Id}, \sigma^2(t)) \quad (15)$$

где $d(t)$ - ‘коэффициент дрейфа’, непрерывная функция, удовлетворяющая $d(0) = 1$ и $d(1) = 0$. $\sigma(t)$ - график шума, $\sigma(0) \rightarrow 0$.

Здесь и далее $*$ обозначает композицию векторов из $\mathfrak{so}(3)$. Для \mathbb{R}^N , имея прямой процесс в виде стохастического дифференциального уравнения, можно получить обратный в таком же виде [10]. Для $SO(3)$ однако это проделать затруднительно, поэтому мы обратились к версии динамики Ланжевена с отжигом (рис. 4), описанному в [4]. Чтобы иметь возможность использовать динамику Ланжевена в обратном процессе, нам необходимо знать распределение $p_t(\mathbf{R}_t|\mathbf{R}_0)$, которое предоставляется теоремой 4.2.

Theorem 4.2. *Для прямого процесса 15:*

$$p_t(\mathbf{R}_t|\mathbf{R}_0) = \text{IGSO}_3(\mathbf{R}_t | \exp[d(t)\mathbf{r}_0], \sigma^2(t)), \quad (16)$$

в частности $p_1(\mathbf{R}_1|\mathbf{R}_0) = \text{IGSO}_3(\mathbf{R}_1 | \mathbf{Id}, \sigma^2(1))$.

Доказательство. Используя то, что $p(\tilde{\mathbf{r}}(t)) = \text{IGSO}_3(\mathbf{Id}, \sigma^2(t))$ и свойство распределения $\text{IGSO}_3 \forall \mathbf{r} \in \mathfrak{so}(3)$:

$$p(\mathbf{r} * \tilde{\mathbf{r}}(t)) = \text{IGSO}_3(\mathbf{r}, \sigma^2(t)). \quad (17)$$

Следственно

$$p(d(t)\mathbf{r}_0 * \tilde{\mathbf{r}}(t)) = \text{IGSO}_3(d(t)\mathbf{r}_0, \sigma^2(t)) \quad (18)$$

и

$$p_t(\mathbf{R}_t|\mathbf{R}_0) = \text{IGSO}_3(\mathbf{R}_t | \exp[d(t)\mathbf{r}_0], \sigma^2(t)). \quad (19)$$

Учитывая то, что $d(1) = 0$ и $0\mathbf{r} = \mathbf{Id}$, получаем что $p_1(\mathbf{R}_1|\mathbf{R}_0) = \text{IGSO}_3(\mathbf{R}_1 | \mathbf{Id}, \sigma^2(1))$. \square

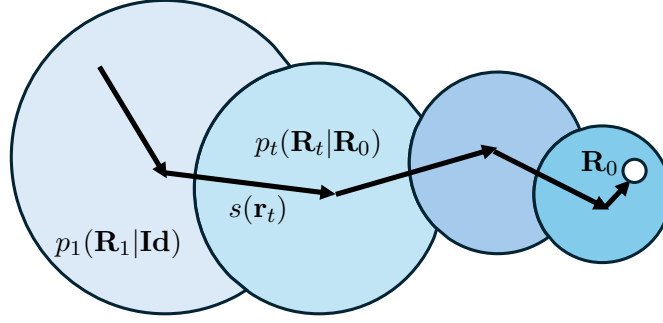


Рис. 1: Динамика Ланжевена с отжигом. Динамика Ланжевена выполняется последовательно для каждого распределения, полученного в результате прямого процесса. На практике обычно одного шага динамики достаточно для каждого распределения.

С учетом данной теоремы, для обратного процесса мы выдвигаем следующее предложение:

Proposition 4.3. *Обратный предложению 4.1 процесс на $SO(3)$:*

$$\mathbf{R}_{t-dt} = \exp[\mathbf{r}_t * g^2(t)s(\mathbf{r}_t)dt * g(t)\sqrt{dt}\tilde{\mathbf{r}}(t)], \quad \tilde{\mathbf{r}}(t) \sim \text{IGSO}_3(\mathbf{Id}, \sigma^2(t)) \quad (20)$$

где $s(\mathbf{r}_t) = \nabla_{\mathbf{r}} \log p_t(\mathbf{r}_t|\mathbf{r}_0)$, $g(t) = \sqrt{\frac{d}{dt}\sigma^2(t)}$ (масштабирование времени).

Доказательство сходимости предложенного процесса целиком для группы $SO(3)$ представляется затруднительным, однако возможно доказать сходимость аналогов динамики Ланжевена:

Lemma 4.4. *При $p(\mathbf{r}|\mathbf{r}_0) = \text{IGSO}_3(\mathbf{r}|\mathbf{r}_0, \sigma^2)$, процесс*

$$\mathbf{r}_{i+1} = \mathbf{r}_i * \epsilon \nabla_{\mathbf{r}} \log p(\mathbf{r}_i|\mathbf{r}_0) * \sqrt{2\epsilon} \mathbf{z}_i, \quad \mathbf{z}_i \sim \text{IGSO}_3(\mathbf{Id}, \sigma^2) \quad (21)$$

сходится к $p(\mathbf{r}|\mathbf{r}_0)$ при $i \rightarrow \infty$.

На практике мы не имеем доступа к истинным значениям градиентов $s(\mathbf{r}_t)$, заменяя их предсказаниями модели $s_\theta(\mathbf{r}_t)$. В этом случае эквивариантность прямого и обратного процесса гарантируется следующей теоремой:

Theorem 4.5. *Прямой процесс 15 эквивариантен относительно $SO(3)$:*

$$\mathbf{R}\mathbf{R}_t(\mathbf{R}_0) = \mathbf{R}_t(\mathbf{R}\mathbf{R}_0) \quad \forall \mathbf{R} \in SO(3). \quad (22)$$

Если $s_\theta(\mathbf{r}_t)$ инвариантен относительно $SO(3)$, то обратный процесс 20 эквивариантен:

$$\mathbf{R}\mathbf{R}_t(\mathbf{R}_1) = \mathbf{R}_t(\mathbf{R}\mathbf{R}_1) \quad \forall \mathbf{R} \in SO(3). \quad (23)$$

Доказательство. Используя то, что для $\mathbf{r}_1, \mathbf{r}_2 \in \mathfrak{so}(3)$ $\exp[\mathbf{r}_1] * \exp[\mathbf{r}_2] = \exp[\mathbf{r}_1 * \mathbf{r}_2]$ и ассоциативность $\mathfrak{so}(3)$, для прямого процесса:

$$\mathbf{R}\mathbf{R}_t(\mathbf{R}_0) = \exp[\mathbf{r}] \exp[d(t)\mathbf{r}_0 * \tilde{\mathbf{r}}(t)] = \exp[\mathbf{r} * d(t)\mathbf{r}_0 * \tilde{\mathbf{r}}(t)] = \mathbf{R}_t(\mathbf{R}\mathbf{R}_0) \quad (24)$$

Для обратного процесса:

$$\begin{aligned} \mathbf{R}_{t-dt}(\mathbf{R}\mathbf{R}_t) &= \exp[\mathbf{r} * \mathbf{r}_t * g^2(t)s_\theta(\mathbf{r} * \mathbf{r}_t)dt * g(t)\sqrt{dt}\tilde{\mathbf{r}}(t)] = \\ &= \exp[\mathbf{r} * \mathbf{r}_t * g^2(t)s_\theta(\mathbf{r}_t)dt * g(t)\sqrt{dt}\tilde{\mathbf{r}}(t)] = \mathbf{R}\mathbf{R}_{t-dt}(\mathbf{R}_t). \end{aligned} \quad (25)$$

Далее по индукции получаем эквивариантность обратного процесса. \square

Вследствие теоремы 4.2 градиенты распределений могут быть посчитаны в явном виде:

$$\nabla_{\mathbf{r}} \log p(\mathbf{r}|\mathbf{r}_0) = \frac{\mathbf{r}^{-1} * \mathbf{r}_0}{\omega(\mathbf{r}^{-1} * \mathbf{r}_0)} \left. \frac{\partial_{\omega} f(\omega)}{f(\omega)} \right|_{\omega=\omega(\mathbf{r}^{-1} * \mathbf{r}_0)}, \quad (26)$$

где $\omega(\mathbf{r})$ - угол вращения, соответствующий \mathbf{r} . На практике, модель эффективнее обучается предсказывать значения \mathbf{r}_0 , на основе которых затем рассчитываются градиенты.

Lemma 4.6. *Если модель $f_{\theta}(\mathbf{r}_t)$, предсказывающая \mathbf{r}_0 , эквивариантна относительно $\text{SO}(3)$, то рассчитанные на основе ее предсказаний градиенты $s_{\theta}(\mathbf{r}_t)$ инвариантны относительно $\text{SO}(3)$.*

5 Вычислительный эксперимент

Согласно [8], последствия неполного зашумления данных при обучении достаточно subtilны. Поскольку в прямом процессе, описанном в [5], $\sigma(1) = 1.0$, данный процесс хоть и не сходится к полному шуму, однако итоговое распределение покрывает большую часть пространства $\text{SO}(3)$. Задачей вычислительного эксперимента было найти такое целевое распределение синтетических данных, при котором предвзятость процесса [5] была бы наглядна.

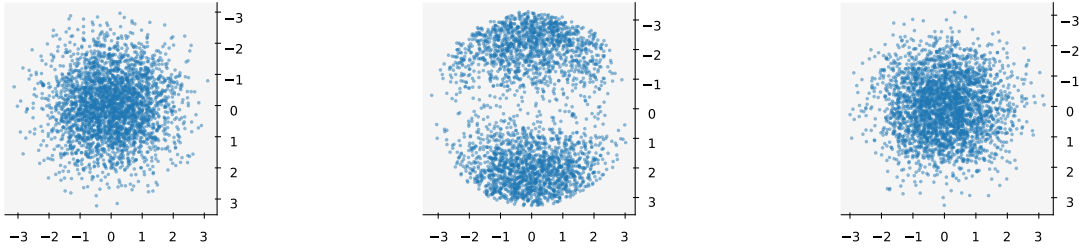


Рис. 2: Слева изображен шум, $\mathbf{r} \sim \text{IGSO}_3(\mathbf{Id}, 1)$. По центру и справа изображены результаты полного зашумления процесса [5] и предлагаемого в данной работе процесса соответственно. Элементы группы $\text{SO}(3)$ изображены в представлении векторов Эйлера и для наглядности спроецированы на плоскость.

На роль такого распределения подходит $p_0(\mathbf{r}) = \mathcal{U}(\mathbf{r}_0, \mathbf{r}_0 + \delta \mathbf{r}_0)$, где $\mathbf{r}_0 = [2, 0, 0]$ и $\delta \mathbf{r}_0 = [0.01, 0.01, 0.01]$ в представлении векторов Эйлера. Результаты прямых процессов для данного распределения изображены на рис. 10. Можно заметить, что результат предлагаемого в данной работе прямого процесса совпадает с ожидаемым распределением $\text{IGSO}_3(\mathbf{Id}, 1)$, в то время как результат процесса, предложенного в [5], значительно от него отличается. В ходе эксперимента для предсказаний использовался Multi-Layer Perceptron (MLP) с 6 слоями и функциями активации ReLU [11]. Размерность латентного пространства модели составила 256, а для кодирования момента времени t были применены синусоидальные кодирования [12]. Функция потерь в процессе обучения определялась как среднеквадратичная ошибка (MSE) на векторах

Эйлера, а оптимизация модели проводилась с использованием алгоритма Adam со стандартными параметрами. Размер батча составил 256, а общее количество точек в датасете - 3000. Обучение модели продолжалось в течение 100 эпох. Сгенерированные данные представляют собой 3000 точек, полученных в результате обратного процесса, состоящего из 500 итераций.

6 Генерация трехмерных структур РНК

Использование реальных данных позволяет оценить применимость и эффективность разработанного метода. Также, проверка на реальной задаче помогает выявить возможные проблемы или ограничения модели, которые могли бы остаться незамеченными при тестировании на синтетических данных. Наконец, проведение эксперимента на реальных данных позволяет более полноценно сравнить результаты с существующими методами и подтвердить превосходство или дополнительные преимущества разработанного нами генеративного процесса. Исходя из имеющихся у нас данных, нами было выбрано применить предложенный процесс к генерации трехмерных структур РНК. В имеющихся у нас данных, молекулы РНК взаимодействуют с молекулами белков. Мы решили учесть это взаимодействие, решая задачу условной генерации из $P(\text{geom}_{RNA} | \text{geom}_{pr}, \text{seq}_{pr}, \text{seq}_{RNA})$, где geom_{RNA} и geom_{pr} - трехмерные структуры РНК и белка, а seq_{pr} и seq_{RNA} - их соответствующие последовательности.

6.1 Представление молекул РНК и белков

Основываясь на предыдущем подходе, использованном в AlphaFold2 (AF2) [13], мы описываем молекулы РНК и белков, используя несколько представлений: одиночное, парное и пространственное (трехмерное, 3D). Основная цель одиночного представления - кодирование информации о каждом остатке в отдельности, в то время как парное представление направлено на захват деталей о взаимодействиях между каждой парой остатков. Мы создаем эти представления, используя информацию о типе остатка, его индексе в последовательности и парных расстояниях.

Мы представляем пространственные трехмерные положения нуклеотидов в РНК и аминокислот в белках в виде жестких рамок, являющимися элементами группы $SE(3)$, которые можно рассматривать как пару из 3D-вектора трансляции и 3D-матрицы вращения. Для рамок аминокислот мы адаптировали стратегию AF2 и построили рамки соответствующим образом, центрируя их на атомах $C\alpha$. Нуклеотиды РНК более громоздки по сравнению с аминокислотами. Поэтому мы решили представить каждый из них двумя отдельными рамками: одна, соответствующая пентозному сахару, и другая - азотистому основанию, как показано на рис. 3.

Рамка сахара центрируется на атомах $C3'$, с Ox в направлении $C2'$ и Oy так, что $C4'$ имеет положительную ординату. Из этой рамки, с помощью пяти дополнительных двугранных углов, указанных на рис. 3, мы можем восстановить все остальные атомы сахара и фосфатной группы. Рамка основания зависит от типа нуклеотида: аденин (A) и гуанин (G) имеют рамку, центрированную на N9, с Ox и Oy , построенными по направлениям N3 и N7 соответственно; урацил (U) и цитозин (C) имеют рамку, центрированную на N1, с Ox и Oy , построенными по направлениям N3 и O2 соответственно. Эти рамки позволяют восстановить другие атомы азотистого основания без

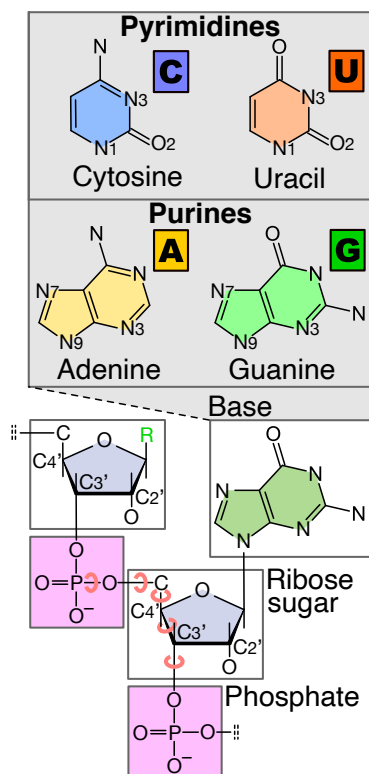


Рис. 3: Схематическое изображение молекулы РНК, состоящей из фосфатной группы, рибозы и основания. Для представления рибозы мы построили жесткие рамки из атомов C3', C2' и C4' (см. основной текст). Для жестких рамок оснований мы используем атомы N9, N3 и N7 в пуринах, и атомы N1, O2 и N3 в пиримидинах. Atom Restoration Network дополнительно предсказывает пять двугранных углов, обозначенных дугами.

дополнительной информации.

6.2 MolBindDif

В данной работе мы исследуем полностью геометрический подход к предсказанию структуры белков и РНК. В частности, в нашей предсказывающей положения атомов (из которых затем по формуле 26 могут быть рассчитаны градиенты плотности распределения) модели MolBindDif мы объединили Invariant Point Attention (IPA) [13] и Axial Attention [14].

MolBindDif состоит из нескольких основных блоков: Encoder, RNA Block, Protein Block и Atom Restoration Network, как показано на рис. 4 и описано ниже. Блоки RNA Block и Protein Block имеют схожую архитектуру (см. рис. 8 и 7 для более подробной информации). Они начинаются с IPA блоков, которые обновляют одиночные представления. Каждому типу взаимодействия (РНК-РНК, РНК-белок, белок-РНК и белок-белок) соответствует отдельный IPA блок, чтобы модель могла проще уловить особенности этих взаимодействий.

Затем мы используем Structure Transition Blocks (архитектура типа ResNet) отдельно для РНК и белков для обработки одиночных представлений, которые затем суммируются с их обновлениями и нормализуются. После этого мы обновляем каждый тип парных представлений с помощью соответствующих обновленных одиночных

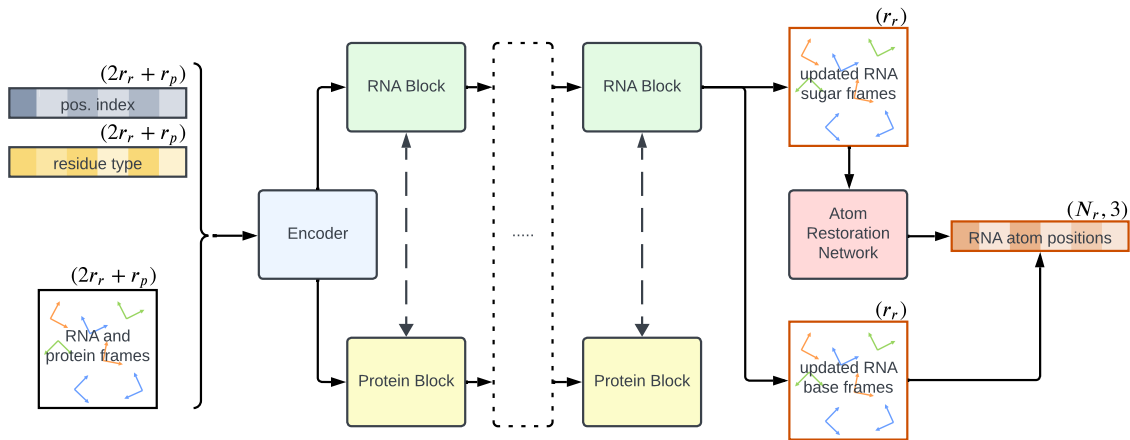


Рис. 4: Схематическое изображение архитектуры MolBindDif. Protein Block и RNA Block показаны более подробно на рисунках 7 и 8 соответственно. Atom Restoration Network показана на рисунке 5. Здесь r_r - количество остатков РНК, r_p - количество остатков белка, а N_r - общее количество атомов в РНК.

представлений, используя Axial Attention (отдельно для каждого типа взаимодействия). В RNA Block мы дополнительно используем Frame Update для обновления рамок РНК. Пара RNA Block и Protein Block может повторяться последовательно (четыре раза в нашей модели). В конце мы напрямую восстанавливаем атомы азотистых оснований из соответствующих рамок. Координаты всех остальных атомов определяются из рамок сахаров с помощью Atom Restoration Network, описанной на рис. 5.

Theorem 6.1. *Архитектура MolBindDif, предсказывающая положения атомов, обладает эквивариантностью относительно $SE(3)$.*

Доказательство. Эквивариантность блоков IPA и Frame Update была доказана в [13]. Их эквивариантность является следствием проведения вычислений в локальных рамках. Остальные блоки модели (Structure Transition, Pairwise Update) инвариантны относительно $SE(3)$, поскольку одиночное представление не зависит от рамок, а парное зависит лишь от относительных расстояния между рамками. В итоге мы получаем общую эквивариантность архитектуры MolBindDif. \square

6.3 Encoder

Для каждого остатка (белка или РНК) сначала кодируется его индекс в последовательности с использованием синусоидных вложений из работы [15], а также временной шаг диффузии с использованием синусоидных вложений из [16]. Типы остатков, специфичные для каждой аминокислоты и сахара/азотистого основания нуклеотида, подвергаются кодированию one-hot, а затем линейному преобразованию. Далее мы используем MLP, чтобы создать одиночные представления, объединяя вложения для индекса, временного шага и типа остатка.

Парные представления строятся аналогичным образом - с помощью MLP (отдельно для каждого типа пары) и объединения вложений. Представление пары РНК-РНК включает относительное кодирование индекса и временного шага, в то время как представления РНК-белок и белок-РНК включают только временные вложения. Для взаимодействий белок-белок мы используем относительный позиционный индекс последовательности и кодирование относительного декартового расстояния. Последнее бинаризуется на 22 интервала с максимальным значением в 20 Å.

6.4 Функция потерь

Обнаружив, что функция потерь, основанная на согласовании градиентов, неэффективна в обучении [5], мы решили реализовать функцию потерь следующим образом:

$$L_{rr} = \sum_i^{N_r} \sum_j^{N_r} (\text{dist}(x_i, x_j) - \text{dist}(x_i^\theta, x_j^\theta))^2, \quad (27)$$

$$L_{rp} = \sum_i^{N_r} \sum_j^{N_p} (\text{dist}(x_i, x_j) - \text{dist}(x_i^\theta, x_j))^2, \quad (28)$$

$$L = L_{rr} + L_{rp}, \quad (29)$$

где L_{rr} и L_{rp} представляют компоненты потерь, связанные с ошибками в относительных положениях РНК-РНК и РНК-белок соответственно. Переменные x_i и x_i^θ обозначают истинные и предсказанные позиции i -го атома. N_r и N_p представляют собой общее количество атомов в молекулах RNA и белка. Предлагаемая функция потерь игнорирует различия между двумя молекулами, если эти различия обусловлены лишь общим вращением или смещением.

6.5 Обучающая, валидационная и тестовая выборки

Мы обучали MolBindDif на комплексах белок-РНК, собранных из базы данных PPI3D [17]. Учитывая потенциальное различие в длине последовательности между полными РНК и их более короткими сайтами взаимодействия с белками, мы решили обрезать РНК последовательности. Для этого мы выбирали взаимодействующие нуклеотиды и их соседей первого и второго порядка в соответствии с диаграммами Вороного, построенными с использованием программного обеспечения VoroContacts [18], которое описывает взаимодействия остатков с помощью тесселяции Вороного атомных шаров. Выбранные нуклеотиды образовали *нуклеотиды связывания* для обучения модели.

Ограниченные памятью графического процессора (GPU), мы уменьшили набор данных, оставив только те комплексы, в которых количество нуклеотидов связывания составляло менее 100, а длина белка - менее 200 аминокислот. Мы использовали одну из предоставленных кластеризаций PPI3D для измерения гомологии белок-РНК комплексов. В этой кластеризации белки в одном кластере имеют менее 40% сходства последовательностей с белками в других кластерах. Для тестового набора данных мы выбрали комплексы, где белок имел менее 40% идентичности последовательности с любой записью в Protein Data Bank (PDB), опубликованной в период до мая 2020 года.

Затем мы добавили в тестовый набор все остальные комплексы из нашего набора данных, которые разделяли кластеры с ранее выбранными. Для обучения мы использовали оставшиеся комплексы, систематически разделяя их в каждой итерации обучения в соотношении 90/10 на тренировочную и валидационную выборку, так что комплексы из одного кластера присутствовали только в одном из наборов.

В конечном итоге тестовая выборка состояла из 1648 образцов, разделенных на 159 кластеров; обучающая и валидационная выборки включали в себя в общей сложности 29 714 образцов, распределенных по 991 кластеру. Количество нуклеотидов связывания было равномерно распределено, среднее значение составляло 48,2 нуклеотида. Большинство белков - более 95% - имели менее 100 аминокислот, с общим средним значением 88,2.

Для устранения дисбаланса данных, где некоторые кластеры более плотно населены, чем другие, и где члены одного кластера могут иметь идентичные структуры, мы приняли стратегию случайной выборки для каждой эпохи обучения. Мы случайным образом выбирали десять образцов для представления каждого кластера (образцы повторяются, если размер кластера меньше 10), обеспечивая более сбалансированное представление данных.

6.6 Детали обучения

Для обучения модели мы использовали оптимизатор Adam [19] с learning rate 10^{-4} , значениями параметров $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-6}$ и параметром `amsgrad = True`. В качестве функции зависимости коэффициента обучения от времени мы использовали ExponentialLR [20] с $\gamma = 0.9$ после каждой 5-й эпохи. Обучение модели проводилось на графическом процессоре A100 с размером мини-батча, равным четырем. Каждый батч содержал один и тот же образец, зашумленный с разными значениями t (значения t выбирались равномерно). Обучение модели продолжалось до тех пор, пока значения функций потерь на обучающем и валидационном наборах не стабилизировались и не перестали уменьшаться, что заняло 529,760 шагов. Важно заметить, что обучение Atom Restoration Network происходило независимо от остальных блоков.

6.7 Atom Restoration Network

Поскольку рамок недостаточно для однозначного восстановления атомной структуры РНК, мы разработали дополнительную модель для этой цели.

Предложенная модель основана на новой концепции, в которой рамки сахара РНК и последовательность нуклеотидов служат входными данными с целью предсказания двугранных углов, необходимых для восстановления положений остальных атомов внутри сахаров и фосфатных групп (см. рис. 3).

Используя структурную информацию, закодированную в рамках сахара и данных о последовательности, модель применяет методы IPA, Axial Attention и AngleResnet. Архитектура схематично представлена на рис. 5.

Мы обучали сеть на том же наборе данных с использованием того же оптимизатора, что и MolBindDif. Однако для функции потерь мы рассматривали только ее L_{rr} компоненту.

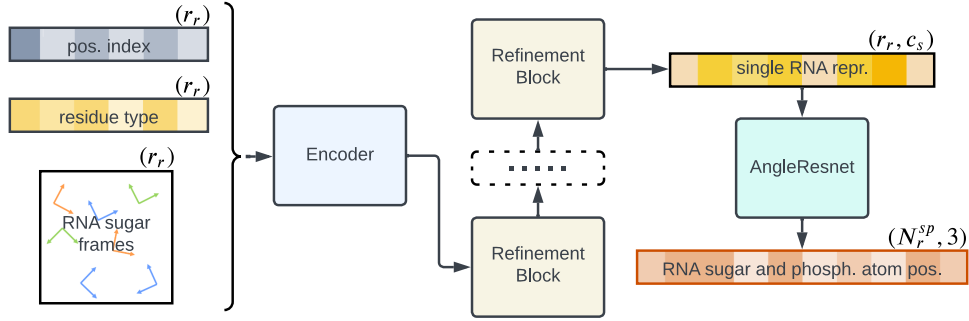


Рис. 5: Схематическое изображение Atom Restoration Network. Refinement Block показан более подробно на рисунке 9. Здесь r_r - количество остатков РНК, c_s - размерность одиночного представления, а N_r^{sp} - общее количество атомов в фосфатных группах и сахарах РНК.

7 Анализ результатов

7.1 Вычислительный эксперимент

В ходе экспериментов было установлено, что предложенный метод превосходит подход, описанный в [5]. Один из основных критериев оценки - это способность метода воспроизводить характер оригинального распределения. Генерируемые методом [5] (см. рис. 6) выборки образуют бимодальное распределение, тогда как исходное распределение таковым не являлось. В то же время предлагаемый нами метод успешно генерировал выборки, которые сохраняли однородность и не имели искусственных мод.

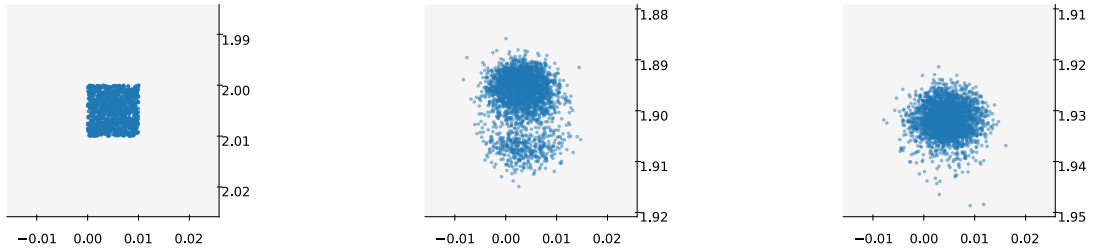


Рис. 6: Слева изображено целевое распределение, по центру - результат генерации метода [5], справа - результат предлагаемого нами метода.

Оба метода продемонстрировали способность генерировать образцы, находящиеся в границах оригинального распределения, что говорит о корректности и стабильности генеративных процессов. Однако, критическое различие заключалось в характере распределения сгенерированных данных. Метод [5] не смог достоверно воспроизвести унимодальное исходное распределение, тогда как новый метод корректно следовал его форме, демонстрируя тем самым свою способность более точно моделировать некоторые распределения.

7.2 Генерация трехмерных структур РНК

Для оценки качества генерации трехмерной структуры молекул с точностью до поворотов и трансляций (применяемых ко всей структуре) мы разработали метрики, вдохновленные IDDT: rRMSD_{rr} , rRMSD_{rp} , IDDT_{rr} и IDDT_{rp} [21]. rRMSD - это средняя ошибка в предсказании расстояний между атомами различных нуклеотидов РНК (rRMSD_{rr}) или между атомами аминокислот белка и нуклеотидов РНК (rRMSD_{rp}). IDDT - это средний процент сохраненных расстояний при заданных пороговых значениях - 0.5 Å, 1.0 Å, 2.0 Å, 4.0 Å. В отличие от оригинальной метрики, мы не использовали обрезание по максимальному расстоянию 15 Å. На практике мы вычисляем IDDT как rRMSD для двух типов расстояний (rr и rp), описанных выше.

Процесс	$\downarrow \text{rRMSD}_{rr}, \text{Å}$	$\downarrow \text{rRMSD}_{rp}, \text{Å}$	$\uparrow \text{IDDT}_{rr}$	$\uparrow \text{IDDT}_{rp}$
Yim et al. 2023 [5]	10.7 ± 5.5	14.1 ± 7.8	0.17 ± 0.06	0.10 ± 0.05
Klypa, 2024	11.1 ± 5.1	13.4 ± 8.2	0.15 ± 0.06	0.11 ± 0.05
RoseTTAFoldNA	13.3 ± 6.3	17.6 ± 7.8	0.19 ± 0.10	0.08 ± 0.04

Таблица 1: Результаты генерации трехмерных структур РНК. RoseTTAFoldNA не является генеративным процессом.

Были сгенерированы (за 100 итераций) трехмерные структуры молекул РНК для 113 различных (меньше 40% сходства последовательностей) белков из тестовой выборки, рассчитанные метрики предоставлены в таблице 1. Также, для наглядности, в таблице представлены результаты негенеративного state-of-the-art метода предсказания трехмерных структур RoseTTAFoldNA [22]. Можно заметить, что различия в результатах между старым и новым генеративными методами незначительны. Оба метода демонстрируют превосходство над RoseTTAFoldNA, что подтверждает сходимость и надежность нового метода при решении реальных задач.

8 Заключение

В данной дипломной работе был предложен новый генеративный процесс согласования градиентов плотности для группы $\text{SO}(3)$. Основное внимание было уделено теоретическим аспектам предложенного метода. Доказанные свойства обеспечивают корректное моделирование и эквивариантность относительно вращений, что критически важно для задач, связанных с трехмерными структурами.

Для оценки эффективности предложенного метода были проведены вычислительные эксперименты на синтетических данных и на реальной задаче условной генерации трехмерных структур молекул РНК. В ходе эксперимента на синтетических данных было продемонстрировано превосходство нового метода в сравнении с существующим подходом [5]: предлагаемый нами метод успешно воспроизвел исходное унимодальное распределение, в то время как старый метод генерировал бимодальное. В задаче условной генерации трехмерных структур молекул РНК оба метода имели схожий уровень производительности, однако различия были замечены в деталях предсказаний. Метод [5] сгенерировал более точно распределение дистанций между РНК и белками, в то время как новый метод показал лучшую точность в генерации взаимодействий РНК-РНК.

Таким образом, предложенный генеративный процесс демонстрирует преимущества в точности и корректности моделирования, особенно в задачах, где критично соблюдение характеристик распределения. Экспериментальные результаты подтверждают потенциал нового метода для применения в различных областях, требующих точного моделирования трехмерных структур, таких как структурная биология и молекулярная динамика.

Список литературы

- [1] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.
- [2] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- [3] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows, 2016.
- [4] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020.
- [5] Jason Yim, Brian L. Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se(3) diffusion model with application to protein backbone generation, 2023.
- [6] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations, February 2021. URL <http://arxiv.org/abs/2011.13456>. arXiv:2011.13456 [cs, stat].
- [7] Dmitry I. Nikolayev and Tatjana I. Savyolov. Normal Distribution on the Rotation Group $So(3)$. *Textures and Microstructures*, 29(3-4):201–233, January 1997. ISSN 0730-3300, 1029-4961. doi: 10.1155/TSM.29.201. URL <https://www.hindawi.com/journals/tsm/1997/173236/abs/>.
- [8] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed, 2024.
- [9] Yesukhei Jagvaral, Francois Lanusse, and Rachel Mandelbaum. Unified framework for diffusion generative models in $so(3)$: applications in computer vision and astrophysics, 2023.
- [10] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.
- [11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.
- [13] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard,

- Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstern, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-021-03819-2. URL <https://www.nature.com/articles/s41586-021-03819-2>.
- [14] Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial Attention in Multidimensional Transformers. 2019. doi: 10.48550/ARXIV.1912.12180. URL <https://arxiv.org/abs/1912.12180>. Publisher: arXiv Version Number: 1.
- [15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. 2017. doi: 10.48550/ARXIV.1706.03762. URL <https://arxiv.org/abs/1706.03762>. Publisher: arXiv Version Number: 7.
- [16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. 2020. doi: 10.48550/ARXIV.2006.11239. URL <https://arxiv.org/abs/2006.11239>. Publisher: arXiv Version Number: 2.
- [17] Justas Dapkunas, Albertas Timinskas, Kliment Olechnovic, Mindaugas Margelevicius, Rytis Diciūnas, and Ceslovas Venclovas. The PPI3D web server for searching, analyzing and modeling protein–protein interactions in the context of 3D structures. *Bioinformatics*, 33(6):935–937, March 2017. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btw756. URL <https://academic.oup.com/bioinformatics/article/33/6/935/2585028>.
- [18] Kliment Olechnovic and Ceslovas Venclovas. VoroContacts: a tool for the analysis of interatomic contacts in macromolecular structures. *Bioinformatics*, 37(24):4873–4875, December 2021. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btab448. URL <https://academic.oup.com/bioinformatics/article/37/24/4873/6300513>.
- [19] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2014. doi: 10.48550/ARXIV.1412.6980. URL <https://arxiv.org/abs/1412.6980>. Publisher: arXiv Version Number: 9.
- [20] Zhiyuan Li and Sanjeev Arora. An Exponential Learning Rate Schedule for Deep Learning. 2019. doi: 10.48550/ARXIV.1910.07454. URL <https://arxiv.org/abs/1910.07454>. Publisher: arXiv Version Number: 3.
- [21] Valerio Mariani, Marco Biasini, Alessandro Barbato, and Torsten Schwede. lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21):2722–2728, November 2013. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btt473. URL <https://academic.oup.com/bioinformatics/article/29/21/2722/195896>.

- [22] Minkyung Baek, Ryan McHugh, Ivan Anishchenko, Hanlun Jiang, David Baker, and Frank DiMaio. Accurate prediction of protein–nucleic acid complexes using RoseTTAFoldNA. *Nature Methods*, 21(1):117–121, January 2024. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-023-02086-5. URL <https://www.nature.com/articles/s41592-023-02086-5>.

Приложение

А Доказательства и выводы

А.1 Вывод формулы 26

Производная по направлению \mathbf{r} от функции $g : (SO)(3) \rightarrow \mathbb{R}^3$ может быть рассчитана следующим образом:

$$D_{\mathbf{r}}g(\mathbf{R}) = \left. \frac{d}{d\epsilon} g(\mathbf{R} \exp[\epsilon \mathbf{r}]) \right|_{\epsilon=0}. \quad (30)$$

Поскольку для IGSO₃ оси вращения распределены равномерно,

$$\nabla_{\mathbf{r}} \log p(\mathbf{r}|\mathbf{r}_0) = \left. \frac{\partial_{\omega} f(\omega)}{f(\omega)} \nabla_{\mathbf{r}} \omega \right|_{\omega=\omega(\mathbf{r}^{-1} * \mathbf{r}_0)} \quad (31)$$

Обозначим ось вращения \mathbf{R} как \hat{l} , ось вращения $\exp[\mathbf{r}]$ как \hat{m} и соответственные углы вращения как α и β . Тогда

$$\omega(\mathbf{R} \exp[\epsilon \mathbf{r}]) = 2 \arccos \left(\cos \frac{\alpha}{2} \cos \frac{\epsilon \beta}{2} - \sin \frac{\alpha}{2} \sin \frac{\epsilon \beta}{2} \hat{l} \cdot \hat{m} \right). \quad (32)$$

Следственно

$$\left. \frac{d}{d\epsilon} \omega(\mathbf{R} \exp[\epsilon \mathbf{r}]) \right|_{\epsilon=0} = \beta \hat{l} \cdot \hat{m}. \quad (33)$$

Таким образом, дифференцируя по направлениям ортогонального базиса $\mathfrak{so}(3)$, мы получаем

$$\nabla_{\mathbf{r}} \log p(\mathbf{r}|\mathbf{r}_0) = \frac{\mathbf{r}^{-1} * \mathbf{r}_0}{\omega(\mathbf{r}^{-1} * \mathbf{r}_0)} \left. \frac{\partial_{\omega} f(\omega)}{f(\omega)} \right|_{\omega=\omega(\mathbf{r}^{-1} * \mathbf{r}_0)}, \quad (34)$$

А.2 Доказательство Леммы 4.4

Для процесса

$$\mathbf{r}_{i+1} = \mathbf{r}_i * \epsilon \nabla_{\mathbf{r}} \log p(\mathbf{r}_i|\mathbf{r}_0) * \sqrt{2\epsilon} \mathbf{z}_i, \quad \mathbf{z}_i \sim \text{IGSO}_3(\mathbf{Id}, \sigma^2) \quad (35)$$

при $p(\mathbf{r}_i|\mathbf{r}_0) = \text{IGSO}_3(\mathbf{r}_i|\mathbf{r}_0, \sigma^2)$ распределение $p(\mathbf{r}_{i+1}|\mathbf{r}_i)$ будет иметь вид

$$p(\mathbf{r}_{i+1}|\mathbf{r}_i) = \text{IGSO}_3 \left(\mathbf{r}_i * \epsilon \tilde{f}(\omega)(\mathbf{r}_i^{-1} * \mathbf{r}_0), \tilde{\sigma}^2 \right), \quad (36)$$

где $\tilde{f}(\omega) = \frac{\partial_{\omega} f(\omega)}{f(\omega)}$. Таким образом, среднее значение распределения $p(\mathbf{r}_{i+1}|\mathbf{r}_i)$ будет всегда находится на геодезической между \mathbf{r}_i и \mathbf{r}_0 (не включая \mathbf{r}_i). Значит для любой сколь угодно малой окрестности \mathbf{r}_0 можно найти такое n , начиная с которого все распределения \mathbf{r}_i будут иметь среднее значение в данной окрестности. Поскольку средние значения стремятся к \mathbf{r}_0 при $i \rightarrow \infty$, процесс сходится к $p(\mathbf{r}|\mathbf{r}_0)$.

А.3 Доказательство Леммы 4.6

Для $(\mathbf{r} * \mathbf{r}_t)^{-1} * f_\theta(\mathbf{r} * \mathbf{r}_t)$ мы имеем:

$$(\mathbf{r} * \mathbf{r}_t)^{-1} * f_\theta(\mathbf{r} * \mathbf{r}_t) = \mathbf{r}_t^{-1} * \mathbf{r}^{-1} * \mathbf{r} * f_\theta(\mathbf{r}_t) = \mathbf{r}_t^{-1} * f_\theta(\mathbf{r}_t). \quad (37)$$

Следственно,

$$s_\theta(\mathbf{r} * \mathbf{r}_t) = \frac{\mathbf{r}_t^{-1} * f_\theta(\mathbf{r}_t)}{\omega(\mathbf{r}_t^{-1} * f_\theta(\mathbf{r}_t))} \frac{\partial_\omega f(\omega)}{f(\omega)} \Big|_{\omega=\omega(\mathbf{r}_t^{-1} * f_\theta(\mathbf{r}_t))} = s_\theta(\mathbf{r}_t) \quad (38)$$

В Дополнительные графические материалы

В.1 Архитектура модели

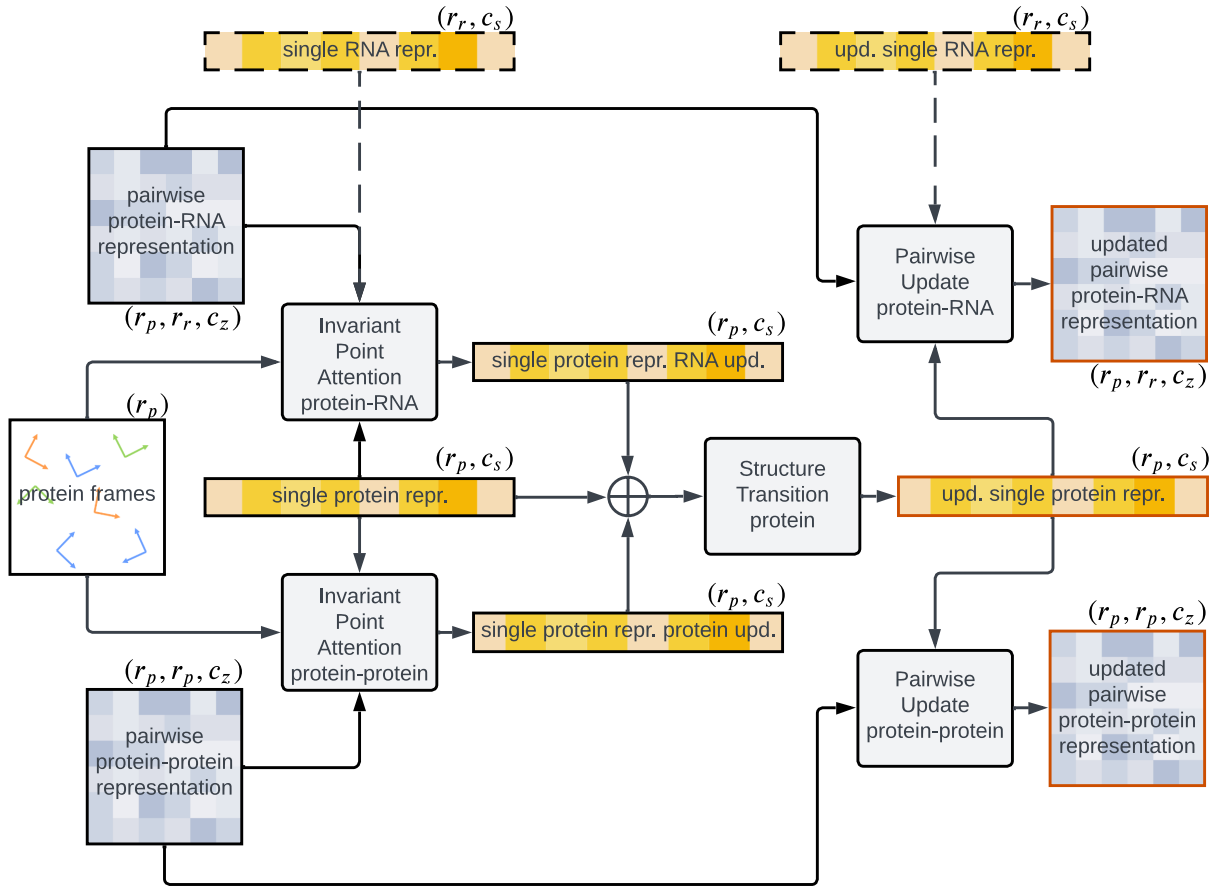


Рис. 7: Архитектура Protein Block. Здесь r_r - количество остатков РНК, r_p - количество остатков белка, c_s - размерность одиночного представления, а c_z - размерность парного представления.

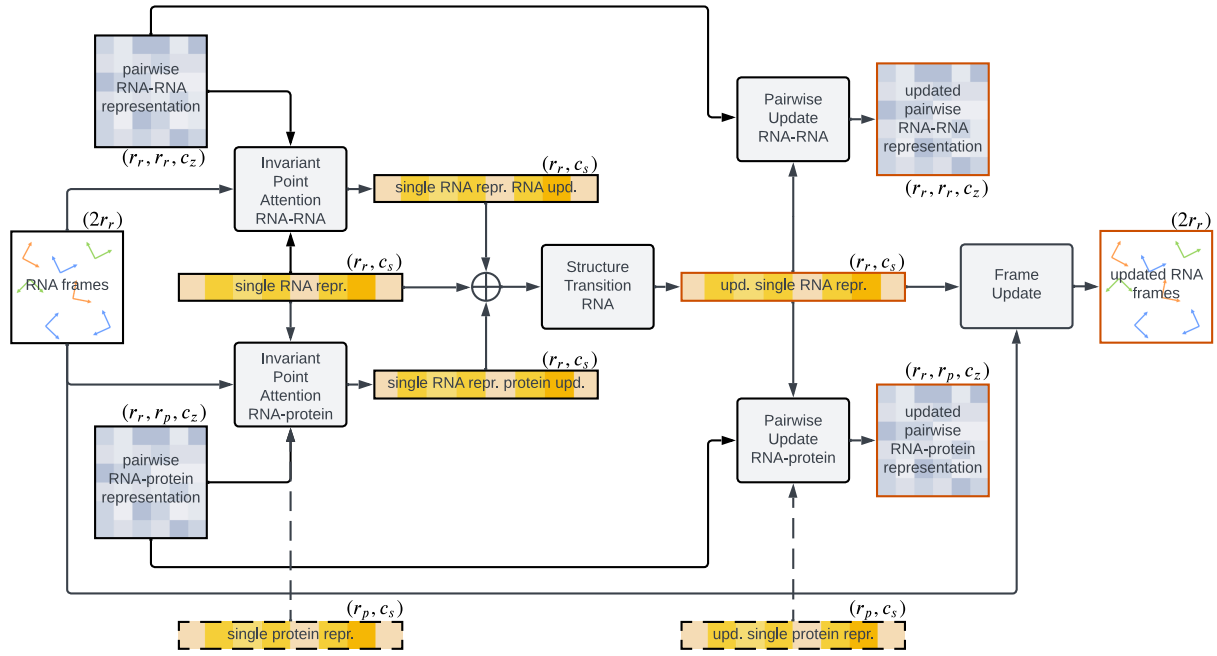


Рис. 8: Архитектура RNA Block. Здесь r_r - количество остатков РНК, r_p - количество остатков белка, c_s - размерность одиночного представления, а c_z - размерность парного представления.

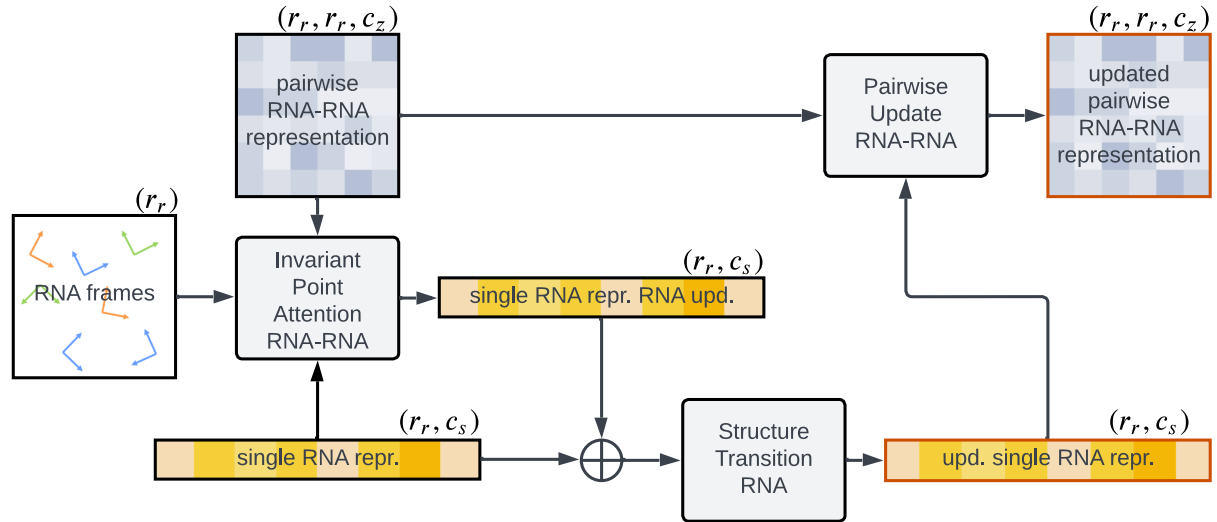


Рис. 9: Архитектура Refinement Block. Здесь r_r количество остатков РНК, c_s размерность одиночного представления, а c_z - размерность парного представления.

В.2 Результат MolBindDif

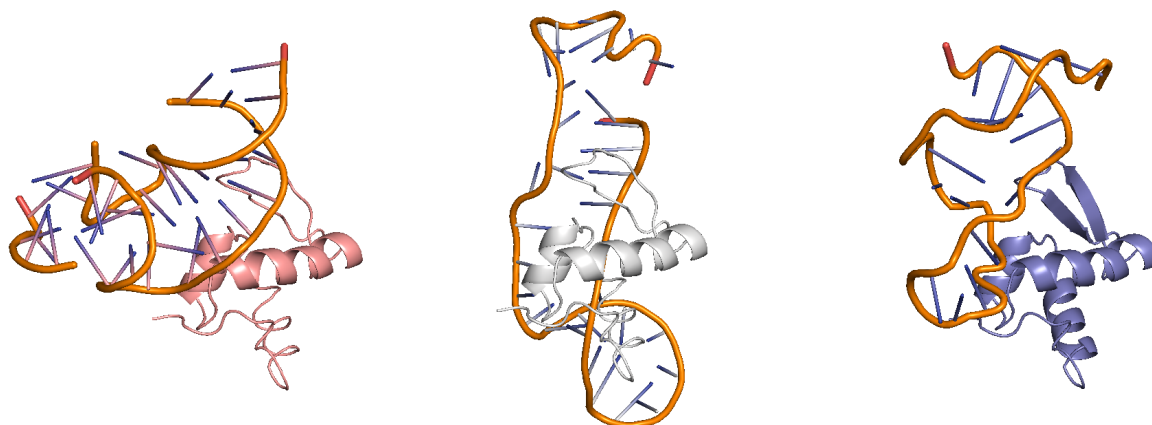


Рис. 10: Слева изображен шум оригинальная молекулярная структура. По центру и справа изображены результаты MolBindDif с генеративным процессом [5] и предлагаемым в данной работе процессом соответственно. Молекулярные структуры были релаксированы с помощью OpenMM, использующим силовое поле Amber14 для улучшения локальной геометрии нуклеотидов.