

1. МАТРИЦЫ КОСИНУСОВ

Матрицы косинусов как математический объект были предложены ранее в рамках курсовых и дипломной работ. Теоретические результаты предыдущих работ здесь будут изложены, по возможности, кратко и без доказательств, но систематично. Здесь они необходимы, прежде всего, для правильного изложения новых результатов, касающихся нового объекта – релятивистской матрицы косинусов. Кроме того, терминология данной главы может местами отличаться от терминологии предыдущих работ, что связано с тем, что она ещё не устоялась, и требует некоторых изменений для большей систематичности.

1.1. Базовые определения

Пусть имеется некоторое гильбертово пространство \mathcal{H} (в частном случае некоторое конечномерное евклидово пространство \mathbb{R}^k). Пусть в данном пространстве задана последовательность точек (ломаная) $X = (\vec{x}_i)$, где $i = \overline{0, n}$. Обозначим $\vec{p}_i = \vec{x}_i - \vec{x}_{i-1}$, где $i = \overline{1, n}$, тогда *ненормализованной матрицей косинусов* по ломаной X будем называть квадратную матрицу вида:

$$C_E = (< \vec{p}_i, \vec{p}_j >)_{ij} \in \mathbb{R}_{n,n}. \quad (1.1)$$

Если выполняется условие:

$$||\vec{p}_i|| = 1, i = \overline{1, n}, \quad (1.2)$$

то говорят о *нормализованных матрицах косинусов* или собственно *матрицах косинусов*. На практике, удобнее всего работать с нормализованными матрицами косинусов, ненормализованные матрицы косинусов могут возникнуть при некоторых операциях над обычными матрицами косинусов. Далее в работе, если не оговорено противное, будет подразумеваться наличие нормализации.

Название матриц следует из того факта, что при условии 1.2:

$$C_E = (< \vec{p}_i, \vec{p}_j >)_{ij} = (||\vec{p}_i|| ||\vec{p}_j|| \cos \vec{p}_i \wedge \vec{p}_j)_{ij} = (\cos \vec{p}_i \wedge \vec{p}_j)_{ij}. \quad (1.3)$$

Объект, определенный выше, строится по и описывает одну ломаную. Пусть имеется две последовательности точек: $X' = (\vec{x}'_i)$, где $i = \overline{0, m}$ и $X'' = (\vec{x}''_j)$,

где $j = \overline{0, n}$. Тогда аналогично 1.1 можно построить матрицу косинусов по двум ломаным X' и X'' :

$$C_{X', X''} = (\langle \vec{p}'_i, \vec{p}''_j \rangle)_{ij} \in \mathbb{R}_{m, n}. \quad (1.4)$$

Определения 1.1 и 1.4 здесь фундаментальны, на практике полезны также матрица косинусов, кодирующая поворот ломаной C_A , и кодирующая вектор относительно ломаной C_ν – они являются частными случаями 1.4. Здесь определим C_ν , в следующей секции – C_A :

Пусть задан некоторый вектор $\vec{\nu} \in \mathbb{R}^k$ и $X = (\vec{x}_i)$ с $i = \overline{0, m}$, тогда *матрицей косинусов задающей вектор* назовём матрицу вида:

$$C_\nu = (\langle \vec{p}_i, \vec{\nu} \rangle)_{ij} \in \mathbb{R}_{m, n}, \quad (1.5)$$

здесь $n \geq 1$ не зависит от ни от X , ни от $\vec{\nu}$, а выбирается из других практических соображений.

Непрерывным обобщение матрицы косинусов можно назвать *поверхностью косинусов*. Пусть $x(\alpha)$ - гладкая кривая $x : [0, l] \rightarrow \mathcal{H}$ и $\vec{p}(\alpha) = \frac{\partial x(\alpha)}{\partial \alpha}$. Тогда 1.1 можно переписать как:

$$C(\alpha_1, \alpha_2) = \langle \vec{p}(\alpha_1), \vec{p}(\alpha_2) \rangle. \quad (1.6)$$

Таким образом, получаем действительную функцию от двух параметров $C : [0, l] \times [0, l] \rightarrow \mathbb{R}$. Условие нормализации для 1.6 выглядит следующим образом:

$$\|\vec{p}(\alpha)\| = 1, \forall \alpha \in [0, l]. \quad (1.7)$$

Использование непрерывной 1.6 для вычислений затруднено. Гораздо удобнее исследовать и использовать матрицы.

1.2. Свойства матриц косинусов

Для исследования свойств описанных матриц, необходимо записать их в более удобном виде. Пусть речь идёт об евклидовом пространстве \mathbb{R}^k , тогда в нём можно ввести репер $O = (\vec{x}_0, \vec{v}_1, \dots, \vec{v}_k)$, порождающий декартову систему координат. Тогда каждой точке \vec{x}_i и вектору \vec{p}_i можно поставить в соответствие координатный вектор-столбец $p_i \in \mathbb{R}_{k, 1}$. Полученные столбцы координат

p_i можно объединить в матрицу $P = [p_1, \dots, p_n] \in \mathbb{R}_{k,n}$. Тогда 1.1 и 1.4 можно переписать как:

$$C_E = P^T P, \quad (1.8)$$

$$C_{X_1, X_2} = P_1^T P_2. \quad (1.9)$$

Теперь, имея некоторую матрицу $A \in \mathbb{R}_{k,k}$ можно определить *матрицу косинусов преобразования* A :

$$C_A = P^T A P. \quad (1.10)$$

1.2.1. Базовые свойства

Приведём основные свойства матриц косинусов (без доказательств):

1. Все значения матриц косинусов лежат на отрезке $[-1, 1]$;
2. Все диагональные элементы матрицы C_E равны 1 (при отсутствии нормализации – $||\vec{p}_i||$);
3. Матрицы косинусов не зависят от выбора декартовой системы координат;
4. Ранг матрицы C_E равен размерности ломаной X ;
5. Матрица C_E - симметричная. Все её собственные значения неотрицательны, и в сумме равны n ;
6. Ломаную X (ломанные X_1 и X_2) можно восстановить из C_E (C_{X_1, X_2}) с точностью до выбора системы координат, причём как для нормированных, так и для ненормированных матриц (в случае последних с C_{X_1, X_2} необходимо знание длин векторов \vec{p}_i).

1.2.2. Восстановление исходных ломаных

Последнее свойство наиболее важно, так как оно позволяет использовать матрицы косинусов как инвариантное представление геометрических ломаных (что крайне полезно при исследовании белковых молекул и их взаимодействий). Зная 1.8 и 1.9 легко построить соответствующие матрицы. Ниже приведём без

доказательств алгоритмы, которые позволяют строить ломаные по этим матрицам:

1. *Восстановление X из C_E* Из описанного выше свойства 5 следует, что матрица C_E будет обладать k неотрицательными собственными значениями $\lambda_1, \dots, \lambda_k$ и собственными векторами $v_1, \dots, v_k \in \mathbb{R}_{n,1}$. Тогда в некоторой системе координат матрица P примет вид:

$$P = \text{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_k})[v_1, \dots, v_k]^T, \quad (1.11)$$

где координаты точек ломаной X восстанавливаются из P через кумулятивную сумму. Формула 1.11 работает и для ненормализованных матриц.

2. *Восстановление X_2 из C_{X_1, X_2} при известном X_1* Введём следующую крайне важную матрицу:

$$T_P = (PP^T)^{-1}P, \quad (1.12)$$

данная матрица (с поправкой на транспонирование) представляет собой матрицу вычисления параметров линейной регрессии. Из 1.9 и 1.12 следует, что:

$$T_{P_1}C_{X_1, X_2} = (P_1P_1^T)^{-1}P_1P_1^TP_2 = P_2, \quad (1.13)$$

данная формула позволяет посчитать P_2 из C_{X_1, X_2} при известном P_1 , из которого, с помощью кумулятивной суммы, можно получить X_2 . Формула 1.13, опять же, работает и для ненормализованных матриц. Также с помощью такого подхода можно восстановить вектор \vec{v} из 1.5.

3. *Восстановление A из C_A при известном X* Данная операция выполняется с помощью двойного применения матрицы 1.12:

$$T_PC_AT_P^T = (PP^T)^{-1}P(P^TAP)P^T(PP^T)^{-1} = A, \quad (1.14)$$

4. *Восстановление X_1 и X_2 из C_{X_1, X_2}* Для выполнения данной операции существует следующий алгоритм, доказательство которого было

приведено в предыдущей работе:

1. Пусть K_1 – матрица собственных векторов $C_{X_1, X_2} C_{X_1, X_2}^T$;
2. Пусть K_2 – матрица собственных векторов $C_{X_1, X_2}^T C_{X_1, X_2}$;
3. $L_1 \leftarrow K_1^T C_{X_1, X_2}$;
4. $L_2 \leftarrow (C_{X_1, X_2} K_2)^T$;
5. $M \leftarrow T_{L_2} C_{X_1, X_2} T_{L_1}^T$;
6. Пусть N – матрица собственных значений $M M^T$;
7. $O \leftarrow N^T M L_2$;
8. $s \leftarrow T_{[O]^2} e$, где e – единичный вектор столбец;
9. $\hat{P}_1 = \text{diag}(\text{sqrt}(s)) O \text{diag}(\|\vec{p}'_1\|, \dots, \|\vec{p}'_m\|)$;
10. $\hat{P}_2 = T_{\hat{P}_1} C_{X_1, X_2}$.

Описанные выше алгоритмы могут работать не только с истинными матрицами косинусов, но и с некоторыми приближёнными/зашумлёнными матрицами. Для этого достаточно производить нормализации по $\|\vec{p}_i\|$ там, где возникают матрицы P . В предыдущих работах было показана устойчивость описанных методов при добавлении случайных шумов.

1.2.3. Арифметические свойства матриц косинусов

Сложение матриц C_E

Пусть имеется две ломаные X' в \mathbb{R}^{k_1} и X'' в \mathbb{R}^{k_2} имеющие соответствующие $P' = [p'_1, \dots, p'_n] \in \mathbb{R}_{k_1, n}$ и $P'' = [p''_1, \dots, p''_n] \in \mathbb{R}_{k_2, n}$. Найдём сумму соответствующих им матриц C'_E и C''_E :

$$C'_E + C''_E = P'^T P' + P''^T P'' = \begin{bmatrix} P' \\ P'' \end{bmatrix}^T \begin{bmatrix} P' \\ P'' \end{bmatrix} = P^+{}^T P^+ = C_E^+, \quad (1.15)$$

где получили новую ломанную $X^+ = X' \oplus X''$ в $\mathbb{R}^{k_1+k_2}$, а также $P^+ = \begin{bmatrix} p'_1 & \dots & p'_n \\ p''_1 & \dots & p''_n \end{bmatrix} \in \mathbb{R}_{k_1+k_2, n}$. Таким образом сумма двух матриц косинусов – ненормализованная матрица косинусов, которой соответствует ломаная, являющаяся прямой суммой ломаных-операнд.

Вычитание матриц C_E

Используя обозначения из 1.15 запишем:

$$C'_E - C''_E = P'^T P' - P''^T P'' = \begin{bmatrix} P' \\ iP'' \end{bmatrix}^T \begin{bmatrix} P' \\ iP'' \end{bmatrix} = P^{-T} P^- = C_E^-. \quad (1.16)$$

Применяя разность в общем случае, придётся выйти из поля действительных чисел, и перейти к полю комплексных чисел. При том, что матрица C_E^- всё еще действительная, порождающая её ломаная $X^- = X' \oplus iX''$ уже принадлежит $\mathbb{C}^{k_1+k_2}$. Для матрицы, полученной из 1.16, в общем случае не будет выполняться свойство о неотрицательности собственных значений. Из-за появления отрицательных собственных значений, при восстановлении X^- из C_E^- необходимо переписать 1.11 как:

$$P^- = \text{diag}(\sqrt{\lambda_1^+}, \dots, \sqrt{\lambda_{k_1}^+}, i\sqrt{-\lambda_1^-}, \dots, i\sqrt{-\lambda_{k_2}^-}) [v_1^+, \dots, v_{k_1}^+, v_1^-, \dots, v_{k_2}^-]^T. \quad (1.17)$$

Произведение матриц косинусов

Ранее было показано, что произведение матриц косинусов может быть использовано для некоторых приближённых вычислений. Причины такого поведения, а также исследование точности приближений здесь опустим, приведём лишь сами приближённые выражения.

Пусть $C_{X_1, X_2} \in \mathbb{R}_{m,n}$ и $C_{X_2, X_3} \in \mathbb{R}_{n,l}$, тогда:

$$C_{X_1, X_2} C_{X_2, X_3} \approx \frac{n}{k} C_{X_1, X_3}. \quad (1.18)$$

Выражение 1.18 приводит к следующим выражениям:

- $C_A C_B \approx \frac{n}{k} C_{AB}$;
- Для ортогональных $A^T = A^{-1}$ выполняется $C_A^T C_A \approx C_A C_A^T \approx \frac{n}{k} C_E$;
- $C_E C_{X_1, X_2} \approx \frac{m}{k} C_{X_1, X_2}$;
- $C_E^2 \approx \frac{n}{k} C_E$.

1.3. Релятивистская матрица косинусов

В данном разделе представлен полностью новый результат, касающийся матриц косинусов. *Релятивистская матрица косинусов* C_{rel} отличается от обычной C_E тем, что позволяет закодировать не только лишь некоторую последовательность точек, но точку, из которой эта последовательность наблюдается. Таким образом получается инвариантное относительно выбора декартовой системы координат (а вообще говоря – Лоренц-инвариантное) представление пары точка-ломаная, что позволяет задавать функции (поля), порождаемые ломаной (белковой молекулой), как преобразования матриц косинусов. Для релятивистских матриц будет дано полное определение, в соответствии со специальной теорией относительности (СТО), и приближённая запись, используемая на практике.

1.3.1. Полное определение

Пусть существует некоторая последовательность точек X , точки которой перемещаются в пространстве \mathbb{R}^k со скоростями, не превышающими скорость света c :

$$\begin{aligned} X(t) &= (\vec{x}_l(t)), l = \overline{0, n}; \\ \vec{x}_l &: \mathbb{R} \rightarrow \mathbb{R}^k; \\ \left\| \frac{\partial \vec{x}_l}{\partial t} \right\| &< c. \end{aligned} \tag{1.19}$$

Следуя Минковскому [ref!], в пространстве-времени можно ввести систему координат, в которой точке \vec{x}_l в момент времени t будет поставлен в соответствие вектор-столбец $x_l = [x_l^1(t), \dots, x_l^k(t), ict]^T \in \mathbb{C}_{k+1,1}$.

Пусть имеется некоторая точка \vec{x}_{ref} и момент времени t_{ref} , из которых наблюдается ломаная X . Этой точке в пространстве-времени соответствуют координаты $x_{ref} = [x_{ref}^1(t), \dots, x_{ref}^k(t), ict_{ref}]^T$. Каждая точка из X будет наблюдаться в \vec{x}_{ref} в момент t_{ref} при пересечении траектории $\vec{x}_l(t)$ и светового конуса, порождённого \vec{x}_{ref} и t_{ref} . Из того, что все скорости не превышают c , следует, что такое пересечение единственно, и для каждой точки \vec{x}_l произойдёт в некоторый момент времени $t_l^* \leq t_{ref}$. Таким образом, в \vec{x}_{ref} в момент t_{ref} будет наблюдаться X^* - образ ломаной $X(t)$ вида: $X^* = (\vec{x}_l^*)$, где $l = \overline{0, n}$ и координаты \vec{x}_l^* равны

$$x_l^* = [x_l^1(t_l^*), \dots, x_l^k(t_l^*), ict_l^*]^T.$$

Если теперь посчитать $\vec{p}_l = \vec{x}_l^* - \vec{x}_{l-1}^*$ и использовать произведение Минковского, 1.1 или 1.8 можно получить *релятивистскую матрицу косинусов*:

$$C_{rel} = (< \vec{p}_l, \vec{p}_j >)_{lj} = \left(\sum_{s=1}^k p_l^s p_j^s - c^2(t_l^* - t_{l-1}^*)(t_j^* - t_{j-1}^*) \right)_{lj} \in \mathbb{R}_{n,n}. \quad (1.20)$$

Отметим, что требование нормализации 1.2 для релятивистских матриц косинусов будет иметь вид:

$$\sum_{s=1}^k (p_l^s)^2 = 1, l = \overline{1, n}, \quad (1.21)$$

т.е. единичную длину должны быть только первые k действительных компонент p_l – мнимая компонента p_l^{k+1} может быть любой.

1.3.2. Упрощённая формулировка

Теперь, для упрощения, положим, что $\forall l : \vec{x}_l(t) = \vec{x}_l = \text{const}$. Также, не нарушая общности, положим $t_{ref} = 0$. При таких предположениях вычисление пересечения светового конуса и траекторий заметно упростится, и можно будет записать:

$$t_l^* = t_{ref} - \frac{||\vec{x}_l - \vec{x}_{ref}||}{c} = -\frac{1}{c} \sqrt{\sum_{s=1}^k (x_l^s - x_{ref}^s)^2}, \quad (1.22)$$

и на основании 1.22 получим следующие значения координат:

$$x_l^* = \begin{bmatrix} x_l^1 \\ \dots \\ x_l^k \\ -i\sqrt{\sum_{s=1}^k (x_l^s - x_{ref}^s)^2} \end{bmatrix}. \quad (1.23)$$

По 1.23 можно посчитать $P^* = \begin{bmatrix} P \\ -id \end{bmatrix} \in \mathbb{C}_{k+1,n}$ и по 1.8 посчитать:

$$C_{rel} = \begin{bmatrix} P \\ -id \end{bmatrix}^T \begin{bmatrix} P \\ -id \end{bmatrix} = P^T P - d^T d = C_E - d^T d \in \mathbb{R}_{n,n}. \quad (1.24)$$

В случае, когда $\|\frac{\partial \vec{x}_l}{\partial t}\| \ll c$ разница между $\vec{x}_l(t_l^*)$ и $\vec{x}_l(t_{ref})$ будет иметь порядок вычислительной погрешности, а следовательно использование 1.20 неоправдано. Далее свойства матриц косинусов будем исследовать исключительно на основе 1.24.

Можно видеть, что по аналогии с обычной C_E можно получить матрицы аналогичные C_{X_1, X_2} , C_A , C_ν :

$$C_{rel, X_1, X_2} = P_1^{*\top} P_2^* = \begin{bmatrix} P_1 \\ -id_1 \end{bmatrix}^\top \begin{bmatrix} P_2 \\ -id_2 \end{bmatrix} = C_{X_1, X_2} - d_1^\top d_2 \in \mathbb{R}_{m, n}, \quad (1.25)$$

$$C_{rel, A} = P_1^{*\top} A P_2^* \in \mathbb{C}_{n, n}. \quad (1.26)$$

Если матрица $A \in \mathbb{C}_{k+1, k+1}$ имеет структуру:

$$A = \begin{bmatrix} A_{11} & iA_{12} \\ iA_{21} & -A_{22} \end{bmatrix}, \quad (1.27)$$

где $A_{11} \in \mathbb{R}_{k, k}$, $A_{12} \in \mathbb{R}_{k, 1}$, $A_{21} \in \mathbb{R}_{1, k}$, $A_{22} \in \mathbb{R}_{1, 1}$; то для 1.26 можно гарантировать:

$$\begin{aligned} C_{rel, A} &= \begin{bmatrix} P \\ -id \end{bmatrix}^\top \begin{bmatrix} A_{11} & iA_{12} \\ iA_{21} & -A_{22} \end{bmatrix} \begin{bmatrix} P \\ -id \end{bmatrix} = \\ &= C_{A_{11}} + d^\top A_{12} P + P^\top A_{21} d + d^\top A_{22} d \in \mathbb{R}_{n, n}. \end{aligned} \quad (1.28)$$

Аналогично, можно построить $C_{rel, \nu}$ и гарантировать, что она будет действительной, если $\nu = [\nu_1, \dots, \nu_k, i\nu_{k+1}]^\top$ ($\nu_l \in \mathbb{R}$).

Матрицы $C_{rel, A}$ и $C_{rel, \nu}$ позволяют кодировать матрицы и векторы в пространстве-времени, также как и C_A , C_ν кодировали матрицы и векторы в евклидовом пространстве. Кроме того, аналогично 1.6 можно ввести релятивистскую поверхность косинусов. Этот объект пригодится для доказательства одного из свойств C_{rel} .

1.3.3. Свойства релятивистских матриц косинусов

Свойства релятивистских матриц аналогичны свойствам обычных матриц:

1. Все значения C_{rel} лежат на отрезке $[-2, 2]$;

2. Все диагональные элементы матрицы C_{rel} лежат на отрезке $[0, 1]$;
3. Релятивистские матрицы косинусов не меняются от преобразований Лоренца (см. 2.2);
4. Матрица C_{rel} - симметричная. Одно её собственное значение отрицательно, остальные – неотрицательны.

Приведём доказательство первых двух свойств, а потом отметим особенности восстановления структур из описанных матриц.

Первое доказательство дадим для непрерывного случая. Пусть $\vec{x}(\alpha)$ - гладкая кривая $\vec{x} : [0, l] \rightarrow \mathbb{R}^k$, и пусть, не нарушая общности, она наблюдается из $x_{ref} = [0, 0, 0, 0]^T$. Тогда координаты вектора $\vec{x}(\alpha)$ будут иметь вид:

$$x(\alpha) = \begin{bmatrix} x_{real}(\alpha) \\ -i||x_{real}(\alpha)|| \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ -i\sqrt{x_1^2 + x_2^2 + x_3^2} \end{bmatrix}, \quad (1.29)$$

и тогда:

$$p(\alpha) = \frac{\partial x(\alpha)}{\partial \alpha} = \begin{bmatrix} \frac{\partial x_1}{\partial \alpha} \\ \frac{\partial x_2}{\partial \alpha} \\ \frac{\partial x_3}{\partial \alpha} \\ -i \frac{x_1 \frac{\partial x_1}{\partial \alpha} + x_2 \frac{\partial x_2}{\partial \alpha} + x_3 \frac{\partial x_3}{\partial \alpha}}{\sqrt{x_1^2 + x_2^2 + x_3^2}} \end{bmatrix} = \begin{bmatrix} \frac{\partial x_{real}(\alpha)}{\partial \alpha} \\ -i \cos \left(x_{real}(\alpha) \wedge \frac{\partial x_{real}(\alpha)}{\partial \alpha} \right) \end{bmatrix}. \quad (1.30)$$

Условие нормализации 1.21 тут имеет вид:

$$||\frac{\partial x_{real}}{\partial \alpha}|| = 1. \quad (1.31)$$

Теперь, аналогично 1.6:

$$\begin{aligned} C_{rel}(\alpha_1, \alpha_2) = < p(\alpha_1), p(\alpha_2) > = \cos \left(\frac{\partial x_{real}(\alpha_1)}{\partial \alpha} \wedge \frac{\partial x_{real}(\alpha_2)}{\partial \alpha} \right) - \\ - \cos \left(x_{real}(\alpha_1) \wedge \frac{\partial x_{real}(\alpha_1)}{\partial \alpha} \right) \cos \left(x_{real}(\alpha_2) \wedge \frac{\partial x_{real}(\alpha_2)}{\partial \alpha} \right). \end{aligned} \quad (1.32)$$

Из 1.32 очевидно следует, что:

$$\forall \alpha_1, \alpha_2 : -2 \leq C_{rel}(\alpha_1, \alpha_2) \leq 2, \quad (1.33)$$

а для случая $\alpha_1 = \alpha_2$:

$$C_{rel}(\alpha, \alpha) = 1 - \cos^2 \left(x_{real}(\alpha) \wedge \frac{\partial x_{real}(\alpha)}{\partial \alpha} \right) = \sin^2 \left(x_{real}(\alpha) \wedge \frac{\partial x_{real}(\alpha)}{\partial \alpha} \right), \quad (1.34)$$

$$\forall \alpha : 0 \leq C_{rel}(\alpha, \alpha) \leq 1. \quad (1.35)$$

Формула 1.32 крайне любопытна, так как она даёт альтернативную аналитическую форму записи поверхности косинусов, имеющую довольно понятное трактование. Предложенное доказательство верно для гладкого случая, однако, если речь идёт о ломаной (и, соответственно, матрице, а не поверхности), следует дать другое доказательство, которое, впрочем, также довольно тривиальное.

Пусть имеется $X = (\vec{x}_l)$, где $l = \overline{0, n}$, которая будет наблюдаться из точки O . Для $\forall l \in \{1, \dots, n\}$ можно обозначить $A = \vec{x}_{l-1}$ и $B = \vec{x}_l$. Пусть $\overline{OA} = d$ и $\overline{OB} = d + \Delta$. Предполагая нормализацию, будем также иметь $\overline{AB} = 1$. По известному свойству треугольников, будем иметь:

$$\begin{aligned} \overline{OB} &\leq \overline{OA} + \overline{AB}, \\ d + \Delta &\leq d + 1, \\ \Delta &\leq 1, \end{aligned} \quad (1.36)$$

и

$$\begin{aligned} \overline{OA} &\leq \overline{OB} + \overline{AB}, \\ d &\leq d + \Delta + 1, \\ -1 &\leq \Delta. \end{aligned} \quad (1.37)$$

Из 1.36 и 1.37 следует $|\Delta| \leq 1$. Тогда C_{rel} запишется как:

$$C_{rel} = \left(\begin{bmatrix} p_l^{real} \\ -i\Delta_l \end{bmatrix}^T \begin{bmatrix} p_j^{real} \\ -i\Delta_j \end{bmatrix} \right)_{lj} = (p_l^{realT} p_j^{real} - \Delta_l \Delta_j)_{lj}, \quad (1.38)$$

откуда аналогично 1.32-1.35 получаем первые 2 свойства.

Восстановление исходных ломаных из релятивистских матриц

Формула 1.14, а также 1.13 (для $C_{rel,\nu}$) применяются здесь так же, как и для обычных матриц косинусов. В то же время применение 1.11 затруднено, и хотя формула 1.17, казалось бы, здесь применима, но наличие элемента $-i\sqrt{\sum_{s=1}^k (x_l^s - x_{ref}^s)^2}$ в 1.23 приводит к тому, что требуются некоторые дополнительные преобразования для согласования координат точек \vec{x}_l и их расстояний до точки наблюдения. Задача несколько упрощается, если известна структура ломаной, и необходимо восстановить только позицию точки наблюдения.

Хотя описанную выше задачу можно сформулировать и решать в терминах задачи оптимизации, эффективного алгоритма её решения пока не предложено. Эксперименты показали, что её можно эффективно решать с помощью нейросетевых алгоритмов.

2. МОДЕЛЬ ВЗАИМОДЕЙСТВИЯ БЕЛКОВ

Цель данной главы – предложить класс физических моделей, которые потенциально могут описывать взаимодействие белковых молекул. В предыдущих работах [!ref] предлагалось напрямую предсказывать конформацию белковых комплексов с помощью полносвёрточных нейронных сетей, и матрицы косинусов использовались в качестве инвариантной кодировки таких конформаций. В данной работе предлагается подход, близкий к классическому методу потенциалов.

2.1. Лагранжев формализм классической релятивистской теории поля

Опираясь на [!ref] кратко изложим суть данного формализма. Пусть имеется четырёхмерное пространство-время (трёхмерное пространство + время). Пусть в данном пространстве можно задать n обобщённых переменных:

$$u_1(x_1, x_2, x_3, x_4), \dots, u_n(x_1, \dots, x_4); \quad (2.1)$$

где $x_4 = ict$. У этих переменных можно вычислить частные производные по каждой из координат: $\frac{\partial u_1}{\partial x_1}, \dots, \frac{\partial u_n}{\partial x_4}$. Также могут использоваться производные более высоких порядков, но в большинстве теорий обходятся лишь первым порядком.

Зададим действительнзначную $L(u_1, \dots, u_n, \frac{\partial u_1}{\partial x_1}, \dots, \frac{\partial u_n}{\partial x_4})$ – *функцию Лагранжа* или *лагранжиан*. В релятивистской теории на лагранжиан накладывается требование *Лоренц-инвариантности*.

Преобразования Лоренца – это такие преобразования, которые сохраняют скалярные произведения в пространстве времени, т.е.:

$$\langle f(x), f(y) \rangle = \langle x, y \rangle, \quad (2.2)$$

при условии, что для пространства-времени скалярное произведение записывается как:

$$\langle x, y \rangle = x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4 = x_1y_1 + x_2y_2 + x_3y_3 - c^2t_xt_y. \quad (2.3)$$

Из курса линейной алгебры известно, что преобразование f , обладающее

свойством 2.2, является изометрическим, и ему будет соответствовать четырёх-мерная комплексная ортогональная матрица A ($A^T = A^{-1}$) (не стоит путать с унитарной матрицей ($\bar{A}^T = A^{-1}$)).

Если обобщённые координаты 2.1 задают скаляры, то их производные будут векторами, если переменные задают векторы, то их производные будут матрицами. При Лоренц-преобразовании, заданном матрицей A ($x \rightarrow Ax$), векторы и матрицы, входящие в лагранжиан, также будут изменяться ($v \rightarrow Av$; $M \rightarrow AMA^T$). Требование Лоренц-инвариантности заключается в том, что преобразования Лоренца не должны изменять значение функции Лагранжа.

Если имеется Лоренц-инвариантный лагранжиан, то можно задать *функцию действия*:

$$S = \int_{spacetime} L(u_1, \dots, u_n, \frac{\partial u_1}{\partial x_1}, \dots, \frac{\partial u_n}{\partial x_4}) dx_1 dx_2 dx_3 dx_4. \quad (2.4)$$

Поведение системы, таким образом, описывается с помощью *принципа стационарного действия*:

$$S \rightarrow \min. \quad (2.5)$$

Задачи типа 2.2 обычно решают с помощью вариационного исчисления. Из принципов вариационного исчисления и инвариантности относительно преобразований 2.2 следует теорема Нётер и автоматически выводимые из неё фундаментальные физические законы сохранения. Поэтому при моделировании белковых взаимодействий, довольно привлекательно выглядит свойство Лоренц-инвариантности, так как с помощью него можно попытаться избежать явного учёта законов сохранения.

2.2. Общая модель взаимодействия белков с применением релятивистских матриц косинусов

Белковые молекулы являются полимерными молекулами. Они являются линейными последовательностями аминокислотных остатков, соединённых пептидными связями. В каждом из остатков находится атом $C\alpha$. В общем случае, знания пространственного расположения этих атомов и знания соответствующих аминокислотных остатков достаточно, чтобы полностью задать структуру белковой молекулы. Кроме того, расстояние между соседними $C\alpha$ -атомами яв-

ляется практически константным, и равно $3,8\text{\AA}$. Эти факты способствуют тому, чтобы можно было использовать матрицы косинусов для задания белковых структур. Введение релятивистской матрицы косинусов 1.20 позволяет задать как структуру молекулы, так и некоторую точку пространства-времени относительно неё. При этом, важным достоинством релятивистской матрицы является её Лоренц-инвариантность. Действительно, легко видеть, что от скалярных произведений, составляющих C_{rel} в 1.20, в 2.2 требуется инвариантность. Таким образом, если обычная матрица косинусов была инвариантна относительно изометрических преобразований в обычном евклидовом пространстве, то релятивистская матрица инвариантна относительно преобразований Лоренца в пространстве-времени. Следовательно, можно задавать обобщенные функции поля как функционалы над C_{rel} . Также необходимо отметить, что, так как C_{rel} содержит в себе закодированным, в том числе, расстояние между точкой наблюдения и наблюдаемыми атомами, но делает это в неявной форме и без обращения этих расстояний, то при использовании описываемого подхода можно обойти проблему бесконечного роста функций поля при сближении атома и точки наблюдения.

Теперь дадим формальное описание данной модели в наиболее общем случае. Для начала, запишем:

$$C_{rel} = C(X, \vec{x}_{ref}) \in \mathbb{R}_{m,m}, \quad (2.6)$$

где X - последовательность точек – $\text{C}\alpha$ -атомов описываемой молекулы в пространстве-времени, \vec{x}_{ref} – точка наблюдения, которой могут соответствовать координаты $x_{ref} = [x_{ref}^1, x_{ref}^2, x_{ref}^3, x_{ref}^4] = [x_{ref}^1, x_{ref}^2, x_{ref}^3, ict_{ref}]$. Тогда 2.1 можно выразить с помощью некоторого набора функционалов, принимающих на вход матрицу и данные об аминокислотной последовательности:

$$u_l = F_l(C_{rel}, A) : \mathbb{R}_{m,m} \oplus \mathbb{A} \rightarrow \mathbb{C}; l = \overline{1..n}. \quad (2.7)$$

Тогда функция действия 2.4 для одной белковой молекулы запишется как:

$$S = \int_{spacetime} L(F_1(C(X, \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}), A), \dots, \frac{\partial F_1(C(X, x), A)}{\partial x_1}, \dots) dx_1 dx_2 dx_3 dx_4. \quad (2.8)$$

Если 2.4 параметризовывалась обобщенными функциями поля (т.е. решением были значения поля во всех точках пространства времени), то 2.8 параметризуется ломаной $X(t)$ в пространстве-времени. Поэтому принцип стационарного действия 2.2 запишется как:

$$X^{sol} = \operatorname{argmin}_X S. \quad (2.9)$$

Выражения 2.8 и 2.9 описывают одну белковую молекулу, и, таким образом, задают модель её свёртки (без явного учёта растворителя и внешних сил). Данные выражения легко обобщить на случай взаимодействия двух молекул, и далее на любое число белковых молекул, и даже на лиганды (для них, впрочем, в общем случае подход с релятивистской матрицей косинусов и функционалом не подходит из-за нелинейности). Запишем модель для двух белковых молекул, описываемых парами (X_1, A_1) и (X_2, A_2) :

$$u_l(x) = F_l(C(X_1, x), A_1) + F_l(C(X_2, x), A_2), l = \overline{1..n}; \quad (2.10)$$

$$S = \int_{spacetime} L(F_1(C(X_1, x), A_1) + F_1(C(X_2, x), A_2), \dots, \frac{\partial F_1(C(X_1, x), A_1)}{\partial x_1} + \frac{\partial F_1(C(X_2, x), A_2)}{\partial x_1}, \dots) dx_1 dx_2 dx_3 dx_4; \quad (2.11)$$

$$(X_1^{sol}, X_2^{sol}) = \operatorname{argmin}_{X_1, X_2} S. \quad (2.12)$$

Таким образом получили наиболее общую модель описания взаимодействия. При отсутствии явных выражений для 2.7 аналитическое исследование такой системы невозможно. Впрочем, даже при наличии таких выражений, оно скорее всего было бы крайне сложным. Численный анализ и поиск таких функци-

оналов также крайне затруднён, особенно с учётом того, что модель описывает полную динамику белковой молекулы, а экспериментальных данных такого рода де-факто не существует. Экспериментально получены лишь статические модели белковых молекул, а динамику получают с помощью расчётов, полученных из теоретических соображений.

2.3. Упрощённая модель взаимодействия белков

Упростим предложенную модель по следующим пунктам:

1. Так как скорости движения белковых молекул крайне малы, по сравнению со скоростью света, то вместо полной релятивистской матрицы косинусов 1.20 следует использовать 1.24.
2. Так как функционалы 2.7 на практике будут аппроксимациями, полученными с помощью машинного обучения, то использование частных производных от таких аппроксимаций может давать весьма сомнительные результаты (что связано с тем, что оператор дифференцирования не является непрерывным в теории функционального анализа), поэтому опустим все явные вхождения частных производных в лагранжиан.
3. Так как вместе с этим опустится производная по времени, то, тем самым, будет получена статическая модель для белковых взаимодействий.
4. Для снижения количества степеней свободы, можно положить, как часто делают, что большие белковые молекулы жёсткие. Тогда они будут моделироваться как абсолютно твёрдые тела, и будут параметризованы своим поворотом и сдвигом в пространстве.

Применяя данные упрощения, мы более не будем иметь дело с релятивистской моделью, однако упрощённая модель всё еще может считаться некоторым классическим механическим приближением такой общей модели.

Рассмотрим последствия предложенных упрощений:

$$X_1(t) = \text{const}, \quad (2.13)$$

$$X_2(t) = \text{const}, \quad (2.14)$$

$$X_1 = X_1(\alpha), \quad (2.15)$$

$$X_2 = X_2(\alpha), \quad (2.16)$$

где α - обобщённый параметр, кодирующий взаимную конформацию молекул X_1 и X_2 . Выражение 2.7 остаётся без изменений. Функция действия для одной молекулы примет вид:

$$\begin{aligned} S_1(\alpha) &= \int_{\mathbb{R}^3} L(F_1(C(X_1(\alpha), x), A_1), \dots, F_n(C(X_1(\alpha), x), A_1)) dx_1 dx_2 dx_3 = \\ &= \text{const}, \end{aligned} \quad (2.17)$$

что обусловлено тем, что повороты и сдвиги не приведут к изменению суммарной функции действия. Функция действия двух молекул будет иметь вид:

$$\begin{aligned} S(\alpha) &= \int_{\mathbb{R}^3} L(F_1(C(X_1(\alpha), x), A_1) + F_1(C(X_2(\alpha), x), A_2), \dots, \\ &F_n(C(X_1(\alpha), x), A_1) + F_n(C(X_2(\alpha), x), A_2)) dx_1 dx_2 dx_3. \end{aligned} \quad (2.18)$$

Принцип стационарного действия примет вид:

$$\alpha_{sol} = \text{argmin}_{\alpha} S. \quad (2.19)$$

Из-за константности 2.17 можно ввести *частичный лагранжиан*:

$$\begin{aligned} L_{partial}(\alpha; x) &= L(F_1(C(X_1(\alpha), x), A_1) + F_1(C(X_2(\alpha), x), A_2), \dots) - \\ &- L(F_1(C(X_1(\alpha), x), A_1), \dots) - L(F_1(C(X_2(\alpha), x), A_2), \dots) = \\ &= L(\alpha; x) - L_1(\alpha; x) - L_2(\alpha; x), \end{aligned} \quad (2.20)$$

вычислив интеграл по которому, получим:

$$\int_{\mathbb{R}^3} L_{partial}(\alpha; x) dx_1 dx_2 dx_3 = S(\alpha) - \text{const} - \text{const}, \quad (2.21)$$

а следовательно:

$$\alpha_{sol} = \operatorname{argmin}_{\alpha} \int_{\mathbb{R}^3} L_{partial}(\alpha; x) dx_1 dx_2 dx_3 = \operatorname{argmin}_{\alpha} S. \quad (2.22)$$

Использование 2.20 будет полезно с численной точки зрения, а также весьма удобно при определённой степени линейности лагранжианов.

2.4. Возможные варианты выбора функции Лагранжа

2.4.1. Скалярное поле

В случае отсутствия частных производных, лагранжиан для скалярного поля u примет вид:

$$L(u) = au^2, \quad (2.23)$$

где a - некоторая константа. Система ведёт себя принципиально разное при положительных и отрицательных a .

Модель с таким лагранжианом крайне бедна. Несколько обогатить её можно если использовать не одно, а n скалярных полей: $U = [u_1, \dots, u_n]^T$. Тогда имея некоторую постоянную $A \in \mathbb{R}_{n,n}$ запишем:

$$L(U) = U^T AU. \quad (2.24)$$

Таким образом получаем мультискалярное поле, функция Лагранжа которого – квадратичная форма. Мультискалярное поле отличается от векторного поля тем, что оно не меняется при преобразованиях Лоренца (ну или хотя бы при поворотах пространства). Не нарушая общности, для квадратичных форм, можно считать A – симметричной ($A = A^T$). Частичный лагранжиан будет иметь вид:

$$\begin{aligned} L_{partial}(U_1, U_2) &= L(U_1 + U_2) - L(U_1) - L(U_2) = \\ &= (U_1 + U_2)^T A(U_1 + U_2) - U_1^T AU_1 - U_2^T AU_2 = \\ &= U_1^T AU_2 + U_2^T AU_1 = U_1^T (A + A^T) U_2 = \\ &= 2U_1^T AU_2. \end{aligned} \quad (2.25)$$

Таким образом, частичный лагранжиан – билинейная форма.

Матрица A выражает взаимодействие различных полей друг с другом. Так

как она симметричная, её можно диагонализировать (получим диагональную A'), и тем самым получить набор новых полей $U' = [u'_1, \dots, u'_n]^T$, которые не будут взаимодействовать друг с другом, а только с самими собой. Вообще говоря, абсолютные значения элементов A' не важны, важны только их знаки, поэтому можно задать 2.24 как:

$$L(U') = U'^T A' U' = \begin{bmatrix} U^+ \\ U^- \end{bmatrix}^T \begin{bmatrix} E & 0 \\ 0 & -E \end{bmatrix} \begin{bmatrix} U^+ \\ U^- \end{bmatrix}, \quad (2.26)$$

где лишние скалярные поля можно полагать нулевыми.

Функционал, способный задать скалярное поле, можно задать различными способами. Эксперименты показали, что наиболее практичный путь - следующий: имея скаляр u , будем задавать вектор $v = [0, 0, 0, -iu]^T$. Тогда 2.7, с помощью некоторого преобразования G , можно представить так:

$$G : \mathbb{R}_{m,m} \oplus \mathbb{A} \rightarrow \mathbb{R}_{m,m}, \quad (2.27)$$

$$C_{rel,v} = G(C_{rel}, A), \quad (2.28)$$

$$u = F(C_{rel}, A) = \begin{bmatrix} 0 & 0 & 0 & i \end{bmatrix} T_X G(C_{rel}, A) \begin{bmatrix} \frac{1}{m} \\ \dots \\ \frac{1}{m} \end{bmatrix}. \quad (2.29)$$

Такое задание скалярное поля - это частный случай задания векторного поля, где собственно векторная часть нулевая. При таком задании, повороты в пространстве не изменяют поля, но преобразования Лоренца в общем случае могут вернуть векторную компоненту, что, впрочем, не является большой проблемой, так как 2.23 можно переписать как:

$$L(v) = au^2 = a < v, v >. \quad (2.30)$$

2.4.2. Векторное поле

Векторное поле задаётся вектором $v = [u_1, u_2, u_3, u_4]^T = [v_1, v_2, v_3, iv_4]^T$. Такое векторное поле не является чисто векторным, но также содержит в себе и скалярное. Его лагранжиан можно задать с помощью 2.30 (легко видеть, что лагранжиан Лоренц-инвариантен) и аналогично 2.24 обобщить на мультивек-

торный случай. Формулу 2.31 можно переписать так:

$$v = F(C_{rel}, A) = T_X G(C_{rel}, A) \begin{bmatrix} \frac{1}{m} \\ \dots \\ \frac{1}{m} \end{bmatrix}. \quad (2.31)$$

В общем и целом векторные поля, заданные таким образом, мало отличаются от скалярных. Главным отличием здесь является чувствительность к поворотам в пространстве, что добавляет некоторых трудностей при обучении задаче предсказания взаимодействия белков. При этом модель оказывается не на много лучше мультискалярной модели.