МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Университет «Дубна»

Институт системного анализа и управления

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
БАКАЛАВРСКАЯ РАБОТА

Тема: Разработка и оптимизация оценочной функции для учета межмолекулярных взаимодействий в белковых комплексах

Ф.И.О. студента Никулин Даниил Андреевич

Группа 4181 **Направление подготовки** 01.03.02 Прикладная математика и информатика

Направленность (профиль) образовательной программы Математическое моделирование

Выпускающая кафедра распределенных информационно-вычислительных систем

Руководитель работы _____ /ст. преп. Полуян С.В. /

Консультант(ы) _____ /_____/

_____ /_____/

Рецензент _____ /доцент, к.б.н. Белов О.В./

Выпускная квалификационная работа
допущена к защите **«____»_____20___г.**

Заведующий кафедрой _____ /Кореньков В.В./

г. Дубна

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Университет «Дубна»

Институт системного анализа и управления

УТВЕРЖДАЮ
Заведующий кафедрой

_____ /Кореньков В.В. /
(Подпись) (Ф И О)

«_____» _____20___ г.

З а д а н и е

на выпускную квалификационную работу – бакалаврскую работу

Тема Разработка и оптимизация оценочной функции для учета межмолекулярных взаимодействий в белковых комплексах.

Утверждена приказом № _____ от _____

ФИО студента Никулин Даниил Андреевич

Группа 4181 **Направление подготовки** 01.03.02 Прикладная математика и информатика

Направленность (профиль) образовательной программы Математическое моделирование

Выпускающая кафедра распределенных информационно-вычислительных систем

Дата выдачи задания «___» _____ 20___ г.

Дата завершения
выпускной квалификационной работы «___» _____ 20___ г.

г. Дубна

Исходные данные к работе

В рамках практики на третьем курсе совместно с научным руководителем разработана библиотека на языке программирования C++ для полноатомного моделирования белка и выполнения структурных изменений в белковых комплексах. В настоящей работе библиотека будет дополнена оценочной функцией. В работе будет произведена реализация и оптимизация оценочной функции для выполнения численной оценки энергии взаимодействия.

Объект и предмет исследования. Математическая и физическая модель образования устойчивого комплекса белок-пептид является предметом исследования. Объектом исследования является комплекс вида белок-белок.

Целями данной работы являются следующие задачи:

1. Разработка оценочной функции для оценки энергии взаимодействия компонентов в комплексах вида белок-белок.
2. Оптимизация вычислительной сложности разработанной оценочной функции с использованием структуры данных k-d-дерево.
3. Сравнение результатов работы разработанной оценочной функции с существующими оценочными функциями.

Результаты работы:

1. Содержание пояснительной записки (перечень рассматриваемых вопросов)

Выполнена реализация и оптимизация оценочной функции на языке программирования C++. Для тестового набора комплексов приведены результаты численных экспериментов, демонстрирующие оценки энергии взаимодействия, полученные с помощью разработанной функции.

2. Перечень демонстрационных листов

Презентация PowerPoint

Руководитель работы _____ /ст. преп. Полуян С. В./

Задание принял к исполнению _____

(дата)

_____

(подпись студента)

АННОТАЦИЯ

В настоящей работе рассматриваются белковые комплексы вида белок-белок, где в качестве компонент комплекса выступают белки, представленные в полноатомном виде. При моделировании процесса образования устойчивого комплекса компонентами при их нековалентном взаимодействии друг с другом возникает необходимость в вычислении энергии такого взаимодействия. Одним из существующих методов для вычисления энергии взаимодействия является использование оценочной функции, которая для заданной пространственной конфигурации компонент позволяет приближенно оценить искомую энергию. В исследовании приведено описание разработанной оценочной функции, в которой учитываются силы межатомных взаимодействий, представленные эмпирическими потенциалами Кулона и Леннард-Джонса. Молекулы растворителя в явном виде не рассматриваются, для этого энергия сольватации вычисляется в рамках модели неявного растворителя. При помощи k-d-дерева произведена оптимизация этапа поиска взаимодействующих атомов между различными компонентами комплекса. Для тестового набора комплексов приведены результаты применения оценочной функции, которые показывают приемлемый уровень корреляции выполняемых оценок при сравнении с существующими инструментами. В различных численных экспериментах продемонстрированы результаты оптимизации оценочной функции, которые демонстрируют уменьшение времени выполнения оценки энергии взаимодействия.

In this work protein-protein complexes considered in full-atom form and consists of several components. Protein-protein complex component assembly is guided by the establishment of non-covalent interactions. To estimate the strength of such interactions at different steps of binding score functions are usually used. In this study presented score function that estimate energy for the given spatial configuration of the protein subunits in complex. Score function considers interatomic interactions through Coulomb and Lennard-Jones potentials. Solvent molecules are not considered explicitly. The implicit solvent model is used to calculate solvation energy. The k-d-tree was used to optimize the search for interacting atoms between different components of the complex. The result of using score fucntion on the test set of protein-protein complexes is presented. It demonstrates an acceptable level of correlation compared to existing tools. The work presents results of various numerical experiments for a test set of different protein-protein complexes which demonstrates an acceptable decrease in the time of score evaluation in comparision with other score instruments.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Функцию для оценки энергии взаимодействия лиганда с белком в заданной пространственной конфигурации называют «оценочной функцией» (scoring function) [1]. В настоящее время разработано множество оценочных функций, которые подразделяются на группы, исходя из принципов их построения. Например, распространено нестрогое деление на эмпирические, статистические и функции на основе силовых полей [2]. Помимо точности оценки искомой энергии взаимодействия важными критерием выбора оценочной функции является вычислительная сложность процедуры оценки, поэтому при моделировании взаимодействия на больших временных масштабах прибегают к моделям с упрощенным представлением белков [3], а также исключают конформационную подвижность, рассматривая компоненты комплекса как «твёрдые» тела, совершающие в растворителе только поступательные и вращательные движения.

Целями работы являются: разработка, реализация, оптимизация и верификация оценочной функции для учета межмолекулярных взаимодействий в белковых комплекса в рамках библиотеки для полноатомного моделирования белковых комплексов PSM [4] (protein structure modeling). Разработка библиотеки PSM проходит в рамках НИР в университете «Дубна», которая позволяет моделировать процесс образования белкового комплекса с помощью кинетического метода Монте-Карло[5]. В основе метода лежит классическая теория переходного состояния, где в процессе моделирования система движется в сторону наименьшей полной энергии по пути с наименьшими энергетическими барьерами, что позволяет модельной системе на пути к термодинамическому равновесию проходить через последовательность квазиравновесных состояний. На определенном этапе метода требуется выполнить оценку взаимодействия, как представлено на рис. 1.

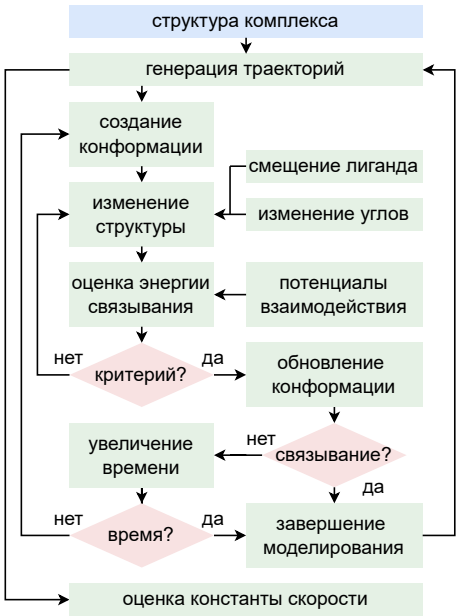

Рис. 1. Основные этапы работы кинетического метода Монте-Карло

При этом, поскольку метод является стохастическим, этап оценки взаимодействия повторяется значительное количество раз. В связи с этим актуальным становится оптимизация этого этапа без существенной потери качества выполняемых оценок. Следует отметить, что для моделирования процесса образования комплекса достаточно сформировать оценочную функцию, учитывающую только парные межатомные взаимодействия и влияние растворителя[3].

Существует множество инструментов для оценки энергии взаимодействия. Например, выполнить оценку взаимодействия возможно с помощью фреймворка Rosetta [6] или силового поля CHARMM [7]. Следует отметить, что для выполнения оценки требуется представление белкового комплекса внутренними средствами инструмента, что, как правило, довольно накладно с временной точки зрения. Например, в случае применения CHARMM перед выполнением оценки требуется перевод структуры во внутренний формат силового поля непосредственно из файла со структурой, а в случае применения Rosetta при выполнении оценки происходит обновление энергетической карты, которое также влияет на время выполнения оценки.

Разработанная в настоящей работе оценочная функция напрямую интегрирована в библиотеку PSM, что позволяет снизить вычислительную сложность выполнения оценки по сравнению с использованием внешних инструментов, что подчеркивает актуальность выполненной работы.

1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Электростатические взаимодействия между мозаично распределенными на поверхности белка электрическими зарядами являются основным фактором, определяющим специфичность взаимодействий в белковых комплексах [8]. Поэтому при разработке оценочной функции для моделирования компонент в виде «твёрдых» тел без конформационных изменений возможно опустить оценку ковалентных и внутримолекулярных нековалетных взаимодействий.

В исследовании для описания межмолекулярных взаимодействий использовались общепринятые эмпирические парные потенциалы Леннард–Джонса и Кулона. Энергия растворителя вычислялась в рамках модели неявного растворителя EEF1 [9]. Процедура оценки энергии с использованием оценочной функции разделяется на два этапа.

На первом этапе формируется список взаимодействующих пар атомов в зависимости от заданного радиуса сферы взаимодействия между атомами указанных компонент белкового комплекса. Для того, чтобы сформировать такой список возможно использовать прямой попарный перебор всех атомов с определением евклидова расстояния. Очевидно, что такой подход очень затратен с вычислительной точки зрения, поскольку каждый компонент комплекса состоит из нескольких тысяч атомов. Указанный этап оптимизируют с использование различных структур данных. Описание выбранной структуры для оптимизации времени поиска атомов приведено в разделе 1.2.

На втором этапе для каждой пары взаимодействующих атомов выполняется оценка энергии взаимодействия, а получившиеся значения суммируются формируя общую оценку взаимодействия для компонент белкового комплекса. Существуют различные подходы для оптимизации указанного этапа, однако они выходят за рамки выполняемой работы.

Разработанная оценочная функция состоит из трёх слагаемых:

$$F_s = E_v + E_c + E_s, \tag{1.1}$$

где E_v представляет собой оценку энергии, полученную с помощью классического потенциала Леннард–Джонса «6–12», слагаемое E_c является оценкой энергии, полученной с помощью потенциала Кулона, E_s – значение оценки энергии растворителя. В следующем разделе для каждого компонента оценочной функции приведено описание с указанием теоретических основ деталей применения.

Поскольку разработка оценочной функции невозможна без учёта физических параметров атомов белка и растворителя в работе использовались данные широко распространённого силового поля CHARMM36 [7]. Принципы применения CHARMM и описание используемых файлов силового поля для выполнения оценок представлены в разделах ниже.

1.1. Компоненты оценочной функции

1.1.1. Потенциал Кулона

Электростатический потенциал Кулона описывает взаимодействие двух постоянных точечных зарядов и определяется следующим образом

$$E_c = \sum_{i,j} \left(\frac{1}{4\pi\varepsilon_r} \frac{q_i q_j}{d_{ij}} \left[\frac{d_{ij}^2}{k^2} - \frac{2d_{ij}}{k} + 1 \right] \right), \tag{1.2}$$

где d_{ij} – евклидово расстояние между центрами атомов, q_i и q_j – фиксированные частичные атомные заряды в рассматриваемой паре атомов, ε_r – диэлектрическая константа. Атомные заряды для каждого атома в зависимости от типа получаются с помощью программы PDB2PQR [10]. Вместо первой дроби при вычислениях используется константа, применяемая в силовом поле CHARMM: 332.0716 ккал·Å·e^{-2}/моль. Последний множитель с коэффициентом $k = 14$Å определяет радиус сферы взаимодействия, для $d_{ij} > k$ вычисления не производятся.

Приведенный потенциал Кулона рассматривается не в классическом виде. Как видно в сумме 1.2 используется дополнительный полиномиальный множитель – квадратная функция, которая необходима для сглаживания значений потенциала при использовании коэффициента отсечения. Поскольку использование такого множителя применяется в силовом поле CHARMM принято решение использовать идентичный принцип вычисления потенциала. На рис. 1.1 представлены результаты вычисления потенциала Кулона без дополнительного множителя и с использованием дополнительным множителя.

1.1.2. Потенциал Леннард-Джонса

Потенциал Леннард-Джонса является моделью парного взаимодействия атомов, которая представляет собой математическую функцию описывающую силы притяжения и отталкивания между атомами на основе их расстояния. Потенциал состоит из двух компонентов, которые позволяют смоделировать эффекты притяжения и отталкивания.

Потенциал Леннард-Джонса «6–12» вычисляется по формуле

$$E_v = \sum_{i,j} \left(\varepsilon_{ij} \left[\left(\frac{R_{ij}}{d_{ij}} \right)^{12} - 2 \left(\frac{R_{ij}}{d_{ij}} \right)^{6} \right] \right), \; R_{ij} = \frac{R_i}{2} + \frac{R_j}{2}, \; \varepsilon_{ij} = \sqrt{\varepsilon_i \varepsilon_j}, \tag{1.3}$$

где d_{ij} – евклидово расстояние между центрами атомов, R_i и R_i – расстояния, на которых значение потенциала становится равным нулю, ε_i и ε_j – глубины потенциальных ям. Указанные параметры получены из файла топологии для соответствующих типов атомов в рассматриваемой паре атомов с индексами i и j.

Необходимо отметить, что потенциал Леннард-Джонса в классическом виде 1.3

не используется в силовом поле CHARMM. Для описания сил Ван-дер-Ваальса используется двойной экспоненциальный потенциал [11]. Он позволяет более точно оценить энергию, поскольку использует отдельные функции для оценки эффекта притяжения и отталкивания, а также отдельную процедуру для учета дальнодействующих взаимодействий. Отличия в получаемых оценках продемонстрированы в разделе 2.3.2.

На рис. 1.1 приведены значения потенциала в зависимости от расстояниям между двумя атомами: углерода ($\varepsilon = 0.11, R/2 = 2$) и водорода ($\varepsilon = 0.031, R/2 = 1.25$).

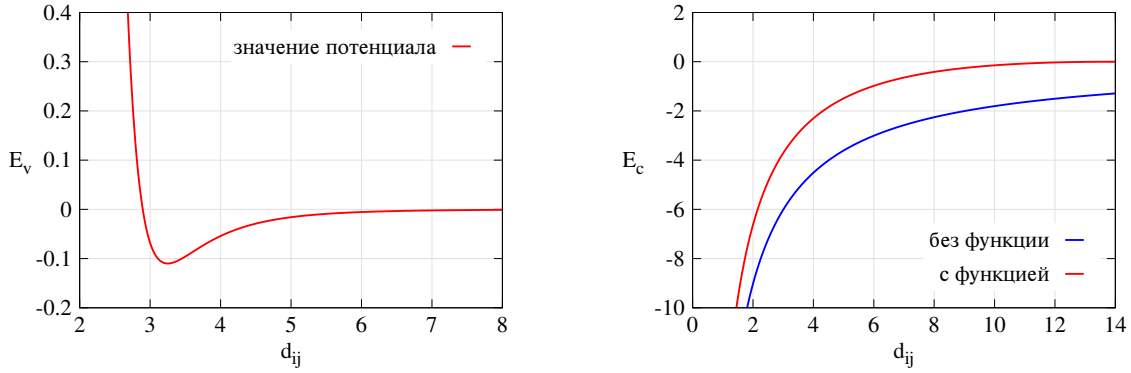

Рис. 1.1. Потенциал Леннард–Джонса. Потенциал Кулона с применением дополнительного множителя и без применения дополнительного множителя

1.1.3. Неявный растворитель

Разрабатываемая оценочная функция в дальнейшем планируется расширить для комплексов вида белок-ДНК. Белковые комплексы, которые могут связываться с фрагментами ДНК присутствуют в большом количестве в клетках организма человека [12]. В настоящей работе рассматриваются глобулярные водорастворимые белки – класс белков, характеризующийся своей структурой и растворимостью в воде. Они имеют компактную, трехмерную структуру, свернутую в форму шара или глобулы. Этот тип белков встречается в цитоплазме клеток и может выполнять разнообразные функции в организме.

Рассматриваемые в работе белковые комплексы проявляют свои нативные свойства лишь в присутствии достаточного количества жидкой воды, которой в цитоплазме приблизительно 80% [13]. Из данных работы [14] следует, что белки обладают гидратной оболочкой, которая представляет собой слой воды, окружающий макромолекулу. Следует отметить, что средняя плотность этой гидратной оболочки на 10% выше, чем плотность объемной воды при аналогичных условиях. Таким образом, на поверхности белков образуется слой связанной воды, физические свойства которой отличаются от свойств воды в объеме.

Взаимодействие с молекулами воды существенно при образовании устойчивого комплекса, поэтому при оценке энергии взаимодействия с помощью оценочной функции необходимо учитывать действие воды. Одним из самых простых способов

моделирования действия водного раствора является неявный растворитель – это математическая модель, используемая для моделирования взаимодействия биомолекул с окружающим растворителем без явного учета молекул растворителя. Такой подход очень эффективен с вычислительной точки зрения, поэтому при разработке было принято решение реализовать модель гауссовских гидратных оболочек EEF1 [9], которая используется для описания структуры гидратации белков.

Гидратная оболочка атома – слой молекул воды, которые окружают атом и связаны с ним посредством водородных связей. Гауссовские функции используются для моделирования гидратационной оболочки вокруг атомов белка, что позволяет более точно выполнить расчет энергии гидратации каждого отдельного атома.

Следует отметить, что в силовом поле CHARMM присутствует возможность оценить энергию с помощью неявного растворителя EFF1, но в работе выполнена собственная реализация модели на основе статьи оригинальной [9] и приведенной в ней таблице значений для типов атомов.

Согласно модели EEF1 энергия растворителя E_s вычисляется для комплекса по формуле:

$$E_s = \sum_i \Delta G_i + \sum_{i,j} \left(\frac{1}{2\pi\sqrt{\pi}} \left[-\Delta G_i e^{-\left(\frac{d_{ij}-R'_i}{\lambda_i}\right)^2} - \Delta G_j e^{-\left(\frac{d_{ij}-R'_j}{\lambda_j}\right)^2} \right] \right), \qquad (1.4)$$

где d_{ij} – евклидово расстояние между центрами атомов, λ_i и λ_j – размеры гидратных оболочек атомов. Параметры R'_i и R'_j определяют Ван-дер-Ваальсовы радиусы атомов, которые соответствуют половине расстояния в потенциале Леннард–Джонса, т.е. $R'_i = R_i/2$, где R_i – параметр, используемый в 1.3. При вычислении первой суммы в выражении 1.4 используются заранее вычисленные при температуре 298.15K значения полной энергии гидратации атомов ΔG_i. Перечисленные выше параметры представлены в таблице [9] и разбиты по группам для каждого типа атома. При реализации оценочной функции использовался принцип определения типов атомов, такой же, как и в силовом поле CHARMM.

1.2. Структура данных k-d-дерево

k-d-дерево – это структура данных, которая позволяет эффективно хранить и обрабатывать точки в многомерном пространстве. Она используется для решения задач, связанных с поиском ближайших соседей, поиском точек в заданном диапазоне и кластеризацией данных.

k-d-дерево представляет собой бинарное дерево, в котором каждый узел соответствует гиперплоскости, разбивающей пространство на две части. Каждый узел содержит точку из множества, которое нужно организовать, а также указатели на двух потомков – левого и правого.

При поиске ближайших соседей или точек в заданном диапазоне, происходит

спуск по дереву, выбирая тот узел, который содержит искомую точку. Затем происходит проверка точек в этом поддереве и, если они удовлетворяют условию поиска, то происходит их добавление в результат. Если же поддерево не содержит искомую точку, происходит переход к следующему поддереву, пока не будет найдена нужная точка или не произведен обход всех поддеревьев.

Для построения k-d-дерева необходимо выбрать гиперплоскость, которая будет разбивать пространство на две части. Это можно сделать различными способами, например, выбрать гиперплоскость, которая проходит через среднюю точку множеств, или выбрать гиперплоскость, которая и в разных поддеревьях.

Структура данных k-d-дерево имеет ряд преимуществ перед другими структурами данных, такими как массивы или хэш-таблицы. Оно позволяет эффективно хранить и обрабатывать большие объемы данных, а также быстро выполнять операции поиска, что важно для поставленной задачи. Применение k-d-дерева является классическим решением задачи оптимизации поиска взаимодействующих атомов и применяется, в том числе, в пакете Rosetta.

1.3. Силовое поле CHARMM

Разработанная оценочная функция использует текстовый файл топологии, который представляет собой набор параметров и топологических данных, используемых для моделирования и симуляции белков с использованием программного пакета CHARMM. Файл $par\_all36m\_prot.prm$ свободно распространяется и требуется для разработанной в настоящей работе оценочной функции, поскольку содержит необходимые параметры для атомов при вычислении оценки потенциалом Леннард-Джонса.

В этом файле содержится информация о параметрах и топологии для атомов, взаимодействий и других особенностей белковой структуры. Несмотря на то, что он также содержит информацию о зарядах атомов, которые получены из экспериментальных данных или рассчитаны с использованием квантово-химических методов, для разработанной оценочной функции информация была получена с использованием программы pdb2pqr [10].

Несмотря на то, что CHARMM открыто распространяется его использование для выполнения оценок затруднительно с вычислительной точки зрения. Применение CHARMM требует преобразования исходного файла PDB во внутреннее представление пакета, которое включает в себя несколько шагов. На рис. 1.2 взятого из [7] представлен путь, который необходимо пройти исходному файлу PDB для получения энергии взаимодействия. Использовать пакет CHARMM напрямую из разрабатываемой в рамках научно-исследовательской работы библиотеки реализованной на C++ затруднительно. Указанные недостатки применения CHARMM легли в основу постановки целей текущей работы.

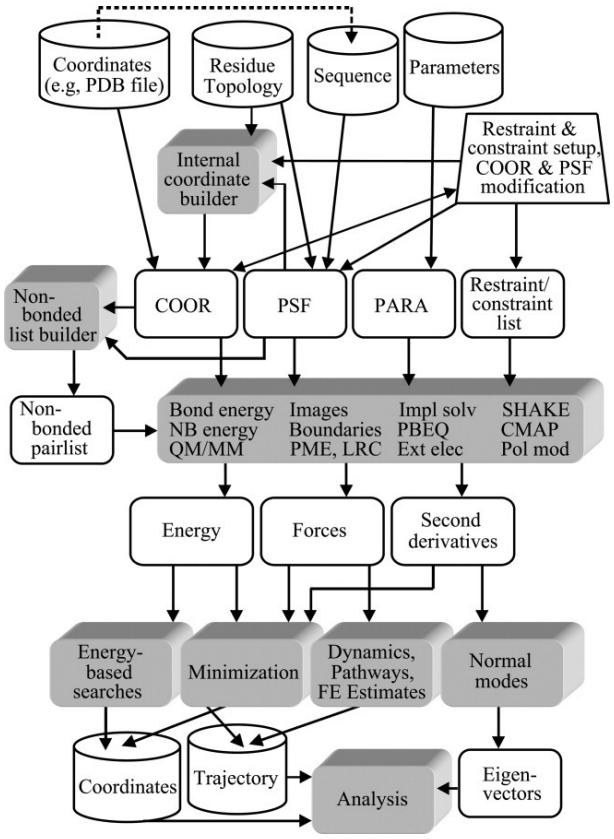

Рис. 1.2. Общая схема проекта CHARMM

1.4. Форматы данных PDB (Protein Data Bank) и PQR

Все рассматриваемые белковые комплексы изначально имеют формат PDB. Архив PDB представляет собой хранилище атомных координат и другой информации, описывающей белки и другие важные биологические макромолекулы. Структурные биологи используют такие методы, как рентгеновская кристаллография, ЯМР-спектроскопия и криоэлектронная микроскопия, чтобы определить положение каждого атома относительно друг друга в молекуле.

Первичная информация, хранящаяся в архиве PDB, состоит из файлов координат биологических молекул. В этих файлах перечислены атомы в каждом белке и их трехмерное расположение в пространстве. Эти файлы доступны в нескольких форматах (PDB, mmCIF, XML). Типичный файл в формате PDB включает в себя большой раздел «заголовка», текста, в котором резюмируется белок, информация о цитировании и детали структурного решения, за которым следует последовательность и длинный список атомов и их координат. Архив также содержит экспериментальные наблюдения, которые используются для определения этих атомных координат.

Уникальным идентификатором каждой структуры в Protein Data Bank является PDB ID. Данный идентификатор состоит из четырех символов, первый из которых, как правило, цифра; буквы в PDB ID принято писать заглавными, хотя большинство программ в данном случае не чувствительны к регистру символов.

В файле данного формата каждая колонка обладает строго определенную длину и содержит определенную информацию: 1-6 – имя записи; 7-11 – серийный номер атома; 13-16 – имя атома; 17 – альтернативное имя атома; 18-20 – имя остатка; 22 – идентификатор цепи; 23-26 – номер остатка; 27 – код для вставки остатков; 31-38 – координата X атома; 39-46 – координата Y атома; 47-54 – координата Z атома; 55-60 – вместимость; 61-66 – температура; 77-78 – символ элемента; 79-80 – заряд атома.

Любые электростатические расчеты начинаются с определения структуры молекулы, параметров заряда и размера составляющих её атомов и свойств растворителя. APBS предоставляет программную утилиту pdb2pqr [10], которая позволяет преобразовать входной файл в формате PDB в формат PQR.

Формат PQR (Protein Data Bank PQR format) является модификацией формата PDB (Protein Data Bank format), используемого для хранения информации о структуре биомолекул, таких как белки, нуклеиновые кислоты и другие макромолекулы. PQR расшифровывается как «PDB with Charges and Radii» (PDB с зарядами и радиусами) и содержит дополнительные сведения о зарядах и радиусах атомов в молекуле.

Основное отличие формата PQR от PDB заключается в наличии дополнительной информации о зарядах и радиусах атомов. В PQR файле каждому атому присваивается заряд (обычно расчетный или эмпирический) и радиус, которые важны для проведения различных анализов и моделирования, включая расчеты электростатических взаимодействий и проницаемости растворителя. Следует отметить, что при преобразовании можно явно указать водородный показатель pH раствора.

2. ПРАКТИЧЕСКАЯ ЧАСТЬ

2.1. Реализация оценочной функции

Оценочная функция была реализована на языке программирования C++ и добавлена в проект PSM в виде набора модулей с соответствующем разбиением на модуль интерфейса и модуль реализации. Для нахождения взаимодействующих атомов был реализован алгоритм прямого перебора всех возможных пар атомов.

Кратко процедура прямого перебора выглядит следующим следующим образом. Обрабатываются списки атомов хранящиеся в ассоциативных контейнерах, где в качестве ключа выступает идентификатор белковой цепи. При рассмотрении каждой пары атомов в контейнере происходит сравнение расстояний между этими атомами и формирование нового списка взаимодействующих атомов.

Очевидно, что в случае, например, фиксированного количества атомов в каждой цепи временная сложность такого алгоритма перебора возрастает со следующей асимптотикой: $O(n^2 \cdot k \cdot (k-1)/2)$, где k – количество цепей, n – количество атомов в одной цепи. В случае различного числа атомов в цепях асимптотика прямого перебора также останется квадратичной.

Для ускорения процесса нахождения взаимодействующих пар атомов использовалась структура данных k-d-дерево. Оптимизированный алгоритм с использованием k-d-дерева работает следующим образом. происходит перебор цепей ассоциативного контейнера и строится соответствующее k-d-дерево. При этом требуется построение $k-1$ деревьев, где k – количество цепей в комплексе. Сложность алгоритма $O(k^2 \cdot n \cdot build)$, где k – количество цепей, n – количество атомов в цепи, по которой будет построено k-d-дерево, а сложность построения дерева $build$ равна $O(n \cdot \log_2 n)$. В общем случае, сложность поиска всех соседей в k-d-дереве составляет $O(\log_2 n + k)$, где n – количество точек в дереве, а k – количество найденных соседей в заданном радиусе сферы взаимодействия атома. Также следует отметить, построенное k-d-дерево обладает фиксированной пространственной сложностью.

В случае отсутствия изменения в позициях атомов некоторых компонентов комплекса нет необходимости строить дерево для каждой оценки. Например, в случае если в комплексе только два компонента и где только один меняет пространственную позицию достаточно построить дерево один раз для неподвижного компонента. В рамках исследования такой подход был реализован и обозначен как k-d-opt.

2.2. Оптимизация с использованием k-d-дерева

В данной работе для ускорения расчетов оценочной функции была реализована структура данных k-d-дерево на языке программирования C++ с поддержкой модулей. Поскольку библиотека PSM для полноатомного моделирования белка реализована на

языке программирования C++ язык реализации оценочной функции и структуры данных k-d-дерева идентичен.

Структура данных представляет собой отдельный подключаемый модуль в рамках библиотеки и представляет собой C++ структуру KDTree, которая хранит в себе указатель на корневой узел дерева и содержит соответствующие методы для построения и поиска взаимодействующих атомов. Листинг кода представлен в Приложении А.

2.3. Результаты численных экспериментов

Выполнение оценок разработанной оценочной функцией возможно только для полноатомных структур, представленных в формате PDB. Поскольку оценочная функция разрабатывалась для моделирования процесса образования комплекса кинетическим методом Монте-Карло, для проверки качества выполняемых оценок был сформирован набор тестовых комплексов, взятый из хранилища SKEMPI [15], представленный в работах [3; 16]. При этом из общего списка комплексов был выделен ограниченный набор структур. Исключены комплексы, содержащие разрывы в главных цепях, а также комплексы, состоящие из трёх и более цепей.

Перед проведением численного эксперимента для всех комплексов проведена предварительная подготовка. С помощью пакета CHARMM выполнено восстановление структур в полноатомный вид, поскольку представленные в базе данных PDB структуры могут не включать определенные атомы. Затем средствами пакета CHARMM выполнена минимизация энергии. В результате для каждого комплекса формируются три стартовых PDB файла, для которых генерируются PQR файлы, содержащие частичные заряды для каждого атома. На последнем этапе производится оценка энергии средствами CHARMM и разработанной оценочной функцией. Сформированный список комплексов и начальных файлов (PDB и PQR) представлен в репозитории [17].

Для проведения верификации оценочной функции использовался облачный сервис университета «Дубна», в рамках которого использовался следующий процессор: Intel Xeon CPU E5-2650. Численные эксперименты, демонстрирующие применение k-d-дерева и сравнение времени поиска взаимодействующих атомов другими инструментами, выполнено на процессоре AMD Ryzen 7 5700X на базовой частоте процессора равной 4.2 ГГц.

2.3.1. Поиск взаимодействующих атомов

На рис. 2.1 представлено сравнение времени поиска взаимодействующих атомов для двух белков различными инструментами.

Тестовый белковые комплексы 1TM1 и 1KXQ выбраны из библиотеки SKEMPI. Их различает разное число атомов. Второй комплекс содержит почти в два раза больше атомов. Комплексы выбраны специально для объективности сравнения оценок при изменении числа атомов. Комплекс 1TM1 имеет два компонента (цепи), которые

содержит 3939 и 1059 атомов соответственно. Комплекс 1KXQ также двухкомпонентный, каждая цепь содержит 7613 и 1786 атомов соответственно. Построение k-d-дерева во всех случаях происходило для первой цепи белкового комплекса.

Как видно из результатов, алгоритм с применением прямого перебора всех атомов работает в несколько раз медленнее других подходов. Однако, при сравнении с Rosetta, несмотря на то, что коэффициент ускорения присутствует и больше единицы, получаемое ускорение незначительно. Это связано с тем, что в Rosetta реализован собственный механизм построения карты взаимодействия атомов, который, в том числе, использует структуру данных k-d-дерево.

Обозначения на рисунках:

1. nn – алгоритм прямого перебора всех атомов;
2. rosetta – программный комплекс Rosetta;
3. k-d – алгоритм с построением k-d-дерева при каждом поиске;
4. k-d-opt – алгоритм с предварительным построением k-d-дерева.

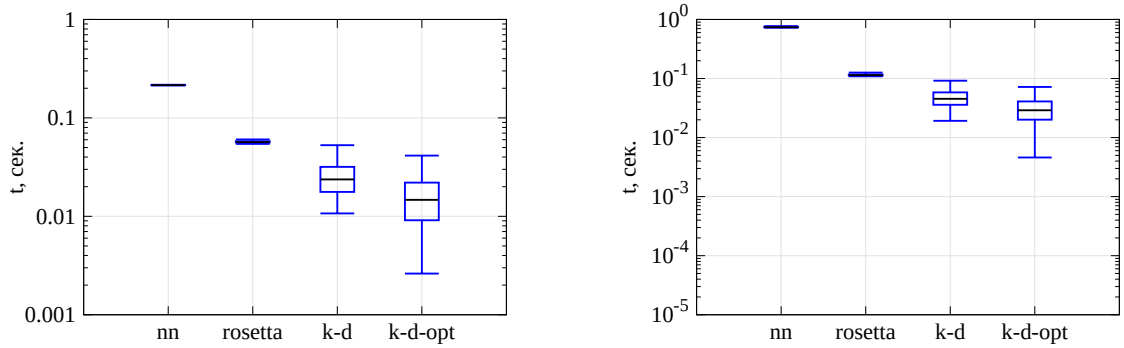

Рис. 2.1. Время поиска взаимодействующих пар атомов для комплексов 1TM1 и 1KXQ

На рис. 2.2 продемонстрированы коэффициенты ускорения соответствующие времени поиска взаимодействующих атомов разработанной функции в отношении с другими алгоритмами. Всего выполнено 9154 оценки для белка 1TM1 и 9479 для комплекса 1KXQ. Следует отметить, что на представленных рисунках число контактов уникально и не повторяется.

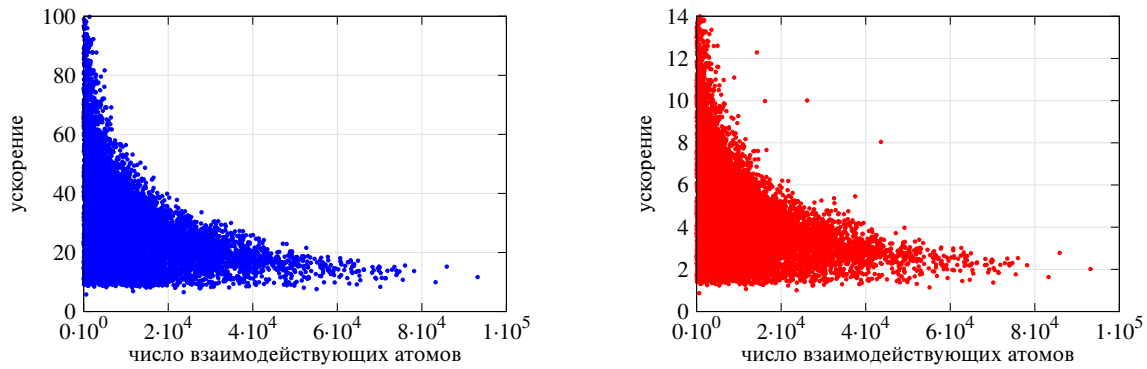

Рис. 2.2. Получаемое ускорение при использовании алгоритма k-d-opt для комплекса 1KXQ по сравнению с алгоритмом прямого перебора nn (слева) и Rosetta (справа)

2.3.2. Верификация выполняемых оценок

На рисунках 2.3 и 2.4 приведены результаты численных оценок для 84 комплексов. Средствами пакета CHARMM выполнена оценка энергии для представленных в разработанной оценочной функции компонент. Для полученных значений рассчитан линейный коэффициент корреляции Пирсона.

Следует отметить, что представленная на рисунках оценка включает в себя внутримолекулярные взаимодействия. Для этого в оценочной функции и в пакете CHARMM использовалась так называемая схема 1-3, где при формировании списка взаимодействующих пар атомов исключаются пары, которые связаны ковалентно (схема 1-2), а также пары, «соединенные» одним общим атомом. При рассмотрении компонент комплекса в виде «твёрдых» тел внутримолекулярные взаимодействия изменяться не будут, поэтому при моделировании процесса образования комплекса достаточно выполнить их оценку только один раз и затем рассматривать взаимодействия только между атомами компонент комплекса.

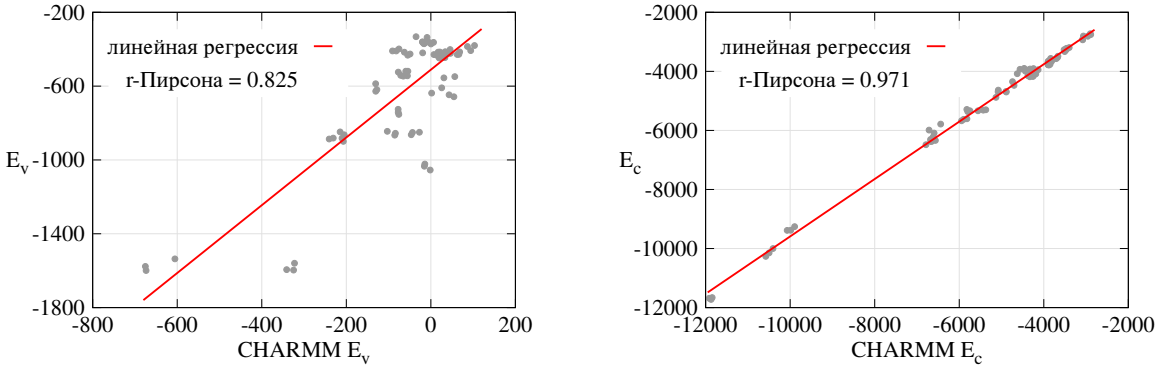

Рис. 2.3. Результаты численного эксперимента для потенциала Леннард-Джонса и для потенциала Кулона

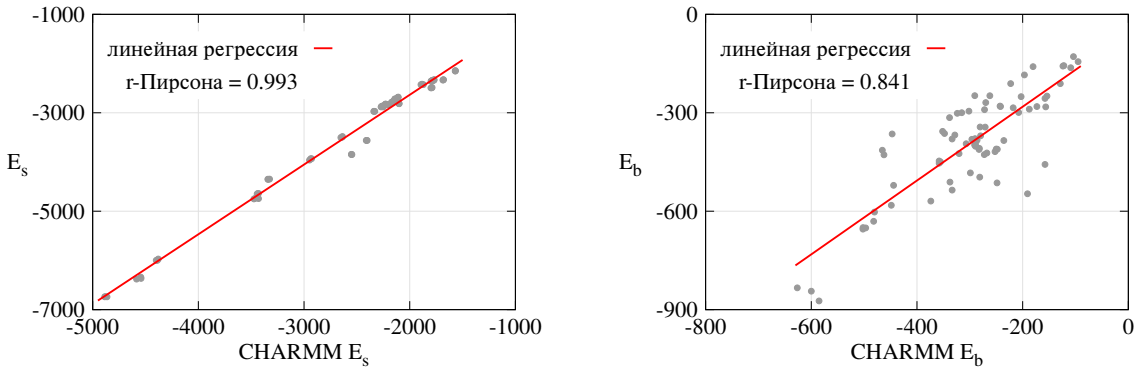

Рис. 2.4. Результаты численного эксперимента для неявного растворителя и для оценки энергии связывания

В простейшем случае процесс связывания описывается моделью вида ключ-замок, которая представляется в виде $A + B \leftrightarrow AB$, где A и B являются компонентами комплекса. С помощью разработанной целевой функции возможно оценить энергию этого взаимодействия. На рисунке 2.4 продемонстрирована оценка энергии связывания

без учета растворителя, которая определяется по следующей формуле:

$$E_b = \left[E_v^{AB} + E_c^{AB}\right] - \left[E_v^A + E_c^A + E_v^B + E_c^B\right]. \tag{2.1}$$

В численном эксперименте в начальном PDB файле представлен образованный комплекс AB. При определении энергии связывания (2.1) вычисляется разница между оценкой энергии комплекса в связанном состоянии и оценками энергий в свободном состоянии для каждого компонента в отдельности. В данном случае оценка энергии растворителя исключена для сравнения результатов оценки взаимодействия только на основе двух слагаемых оценочной функции.

На рис. 2.5 для тестового набора белков продемонстрировано сравнение оценок энергии связывания, которые были вычислены с помощью разработанной оценочной функции (обозначено F_s). Для наглядности посчитана линейная регрессия и коэффициент корреляции Пирсона.

В отличии от оценки E_b, слагаемые которой представлена в выражении 2.1, полная энергия связывания включает в себя все три компонента (включая растворитель) и вычисляется как разница между оценкой энергии комплекса в связанном состоянии и оценками энергий в свободном состоянии, т.е. когда компоненты комплекса друг с другом не взаимодействуют. Представленная на рис. 2.5 оценка энергии связывания $\mathrm{F_s}$ определяется следующим образом:

$$\mathrm{F_s} = F_s^{AB} - \left[F_s^A + F_s^B\right], \tag{2.2}$$

где F_s^{AB}, F_s^A и F_s^B найденные с помощью разработанной оценочной функции 1.1 значения.

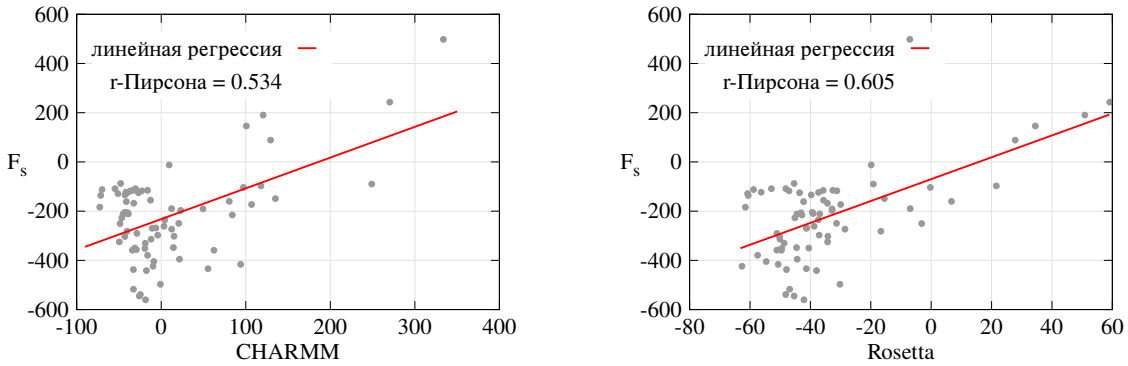

Рис. 2.5. Сравнение оценок энергии связывания разработанной оценочной функции и оценок вычисленных с помощью других инструментов

ЗАКЛЮЧЕНИЕ

В результате выполненной работы достигнуты следующие результаты.

1. Разработана и реализована оценочная функция с использованием набора параметров силового поля CHARMM в рамках библиотеки PSM.
2. Выполнена оптимизация процедуры поиска взаимодействующих пар атомов с помощью применения структуры данных k-d-дерево.
3. Проведены различные численные эксперименты, демонстрирующие приемлемую высокую корреляцию оценок с результатами силовых полей CHARMM и Rosetta.
4. На примере двух белков продемонстрировано преимущество применения структуры данных k-d-дерево для поиска взаимодействующих пар атомов.

Результаты работы представлялись на всероссийской конференции «Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем 2023», которая проходила 17-21 апреля 2023 году в Москве. Тезисы доклада представлены в работе [18].

Задача разработки и реализации оценочной функции поставлена моим научным руководителем, который также является руководителем научно-исследовательской работы проводимой в университете. Разработанная оценочная функция используется в рамках программного пакета PSM, позволяет гибко настраивать все параметры потенциалов и быстро выполнять оценку энергии взаимодействия в комплексах.

Следует отменить, что несмотря на то, что применение структуры данных k-d-дерево позволило в несколько раз уменьшить временную сложность поиска атомов даже в случае большого количества взаимодействующих атомов, время поиска попадает в довольной широкий диапазон значений. Указанное отклонение можно значительно уменьшить с помощью использования других структур данных. Применение других структур данных для поиска взаимодействующих атомов может стать темой дальнейших исследований.

СПИСОК ЛИТЕРАТУРЫ

1. Лизунов А. Ю., Зайцева Н. И., Зосимов В. В. Учет взаимодействий между атомами лиганда в задаче докинга с помощью потенциала усредненных энергий // Труды МФТИ. — 2014. — Т. 6, № 1.
2. Liu J., Wang R. Classification of Current Scoring Functions // Journal of Chemical Information and Modeling. — 2015. — Т. 55, № 3. — DOI: 10.1021/ci500731a.
3. Dhusia K., Su Z., Wu Y. Using Coarse-Grained Simulations to Characterize the Mechanisms of Protein–Protein Association // Biomolecules. — 2020. — Т. 10. — DOI: 10.3390/biom10071056.
4. Репозиторий библиотеки PSM (protein structural modeling). — URL: https://vcs.uni-dubna.ru/PoSV.th/psm (дата обр. 30.05.2023).
5. Voter A. Introduction to the kinetic Monte Carlo method // NATO Science Series. — 2007. — Т. 235. — DOI: 10.1007/978-1-4020-5295-8_1.
6. Alford R. F. [и др.]. The Rosetta all-atom energy function for macromolecular modeling and design // Journal of Chemical Theory and Computation. — 2017. — Т. 13, № 6. — С. 3031—3048. — DOI: 10.1021/acs.jctc.7b00125.
7. Brooks B. [и др.]. CHARMM: The biomolecular simulation program // Journal of Computational Chemistry. — 2009. — Т. 30. — DOI: 10.1002/jcc.21287.
8. Хрущев С. С. [и др.]. Моделирование белок-белковых взаимодействий с применением программного комплекса многочастичной броуновской динамики ProKSim // Компьютерные исследования и моделирование. — 2013. — Т. 1. — DOI: 10.20537/2076-7633-2013-5-1-47-64.
9. Lazaridis T., Karplus M. Effective energy function for proteins in solution. — 1999. — DOI: 10.1002/(SICI)1097-0134(19990501)35:2<133::AID-PROT1>3.0.CO;2-N.
10. Jurrus E. [и др.]. Improvements to the APBS biomolecular solvation software suite // Protein Science. — 2018. — Т. 27. — DOI: 10.1002/pro.3280.
11. Wu X., Brooks B. A double exponential potential for van der Waals interaction // AIP Advances. — 2019. — Т. 9. — DOI: 10.1063/1.5107505.
12. Belov O. V. [и др.]. A quantitative model of the major pathways for radiation-induced DNA double-strand break repair // Journal of Theoretical Biology. — 2015. — DOI: 10.1016/j.jtbi.2014.09.024.
13. Shepher d V. A. The Cytomatrix as a Cooperative System of Macromolecular and Water Networks // Current Topics in Developmental Biology. — 2006. — Т. 75. — DOI: doi.org/10.1016/S0070-2153(06)75006-2.
14. Svergun D. I. [и др.]. Protein hydration in solution: experimental observation by x-ray and neutron scattering // PNAS. — 1998. — DOI: 10.1073/pnas.95.5.2267.

15. Jankauskaite J. [и др.]. SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation // Bioinformatics. — 2019. — Т. 35. — DOI: 10.1093/bioinformatics/bty635.

16. Qin S., Pang X., Zhou H. Automated Prediction of Protein Association Rate Constants // Structure. — 2011. — Т. 19. — DOI: 10.1016/j.str.2011.10.015.

17. Репозиторий со списком комлексов и результатми эксперимента. — URL: https://vcs.uni-dubna.ru/psm/data (дата обр. 30.05.2023).

18. Полуян С. В., Никулин Д. А., Ершов Н. М. Разработка и верификация оценочной функции для учета межмолекулярных взаимодействий в белковых комплексах // Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems 2023 (ITTMM 2023) Book of Abstracts. — 2023.

ПРИЛОЖЕНИЕ А

```
export module kdtree;
/// system
import <vector>;
import <tuple>;
/// internal
import atom;
namespace psm
{
namespace kd
{
 struct KdNode
 {
  psm::Atom* point;
  KdNode* left;
  KdNode* right;
  KdNode() {}
  KdNode (psm::Atom* p)
   : point(p)
   , left(nullptr)
   , right(nullptr) {}
  ~KdNode()
  {
   delete left;
   delete right;
  }
 };
 export struct KDTree
 {
  typedef std::vector
    <std::tuple<psm::Atom*,
                psm::Atom*,
                double,
                double>> atom_pairs_type;
  KdNode* root_;
  KDTree(std::vector<psm::Atom*>& atoms);
  void search(psm::Atom* point,
              const double distance,
                           atom_pairs_type& pairs_data) const;
  KdNode* build(std::vector<psm::Atom*>& points,
                const int depth,
                const int start,
                const int end);
  void search(const KdNode* node,
              psm::Atom* point,
              const double distance,
              const int depth,
              atom_pairs_type& pairs_data) const;
  ~KDTree();
 };
} // kd
} // psm
```

Листинг 5.1. Интерфейс модуля k-d дерево

```
module kdtree;
/// system
import <vector>;
import <tuple>;
import <cmath>;
/// internal
import atom;
namespace psm
{
namespace kd
{
 KDTree::KDTree(std::vector<psm::Atom*>& points)
 {
  root_ = build(points, 0, 0, points.size() - 1);
 }
 KDTree::~KDTree()
 {
  delete root_;
 }
 KdNode* KDTree::build(std::vector<psm::Atom*>& points,
                       const int depth,
                       const int start,
                       const int end)
 {
  if(start > end)
   return nullptr;
  int mid = (start + end) / 2;
  if(depth % 3 == 0)
  {
   std::nth_element(points.begin() + start,
                    points.begin() + mid,
                    points.begin() + end + 1,
   [](psm::Atom* a, psm::Atom* b)
   {
    return a->get_coordinates().x < b->get_coordinates().x;
   });
  }
  else if(depth % 3 == 1)
  {
   std::nth_element(points.begin() + start,
                    points.begin() + mid,
                    points.begin() + end + 1,
   [](psm::Atom* a, psm::Atom* b)
   {
    return a->get_coordinates().y < b->get_coordinates().y;
   });
  }
  else
  {
   std::nth_element(points.begin() + start,
                    points.begin() + mid,
                    points.begin() + end + 1,
   [](psm::Atom* a, psm::Atom* b)
   {
    return a->get_coordinates().z < b->get_coordinates().z;
   });
  }
  KdNode* node = new KdNode();
  node->point = points[mid];
  node->left = build(points, depth + 1, start, mid - 1);
  node->right = build(points, depth + 1, mid + 1, end);
  return node;
 }
 void KDTree::search(const KdNode* node,
                     psm::Atom* point,
                     const double distance,
                     const int depth,
                     atom_pairs_type& pairs_data) const
 {
  if(node == nullptr)
        return;
```

```
  double dd = node->point->get_coordinates()
                    .squared_distance(point->get_coordinates());
  double d = std::sqrt(dd);
  if(distance >= d)
        pairs_data.push_back(
                        std::make_tuple(node->point, point, d, dd));
  if(depth % 3 == 0)
  {
   if(point->get_coordinates().x - distance
     < node->point->get_coordinates().x)
   {
        search(node->left, point, distance, depth + 1, pairs_data);
   }
   if(point->get_coordinates().x + distance
     > node->point->get_coordinates().x)
   {
        search(node->right, point, distance, depth + 1, pairs_data);
   }
  }
  else if(depth % 3 == 1)
  {
   if(point->get_coordinates().y - distance
     < node->point->get_coordinates().y)
   {
    search(node->left, point, distance, depth + 1, pairs_data);
   }
   if(point->get_coordinates().y + distance
     > node->point->get_coordinates().y)
   {
    search(node->right, point, distance, depth + 1, pairs_data);
   }
  }
  else
  {
   if(point->get_coordinates().z - distance
     < node->point->get_coordinates().z)
   {
    search(node->left, point, distance, depth + 1, pairs_data);
   }
   if(point->get_coordinates().z + distance
     > node->point->get_coordinates().z)
   {
    search(node->right, point, distance, depth + 1, pairs_data);
   }
  }
 }
 void KDTree::search(psm::Atom* point,
                     const double distance,
                     atom_pairs_type& pairs_data) const
 {
  search(root_, point, distance, 0, pairs_data);
 }
} // kd
} // psm
```

Листинг 5.2. Реализация модуля k-d дерево

```
export module energy;
/// system
import <iostream>;
import <utility>;
import <string>;
import <vector>;
import <tuple>;
import <map>;
import <unordered_map>;
import <fstream>;
import <chrono>;
/// internal
import atom;
import protein;
import mover;
import residue;
import ienergy;
import kdtree;
namespace psm
{
 // ...
 export class KdScoreFunction : public psm::ScoreFunction
 {
 public:
  using ScoreFunction::init;
  using ScoreFunction::get_atom_pairs_size;
  void exclusion_nbond() override
  {
   pairs_data.clear();
   for (auto iter = atoms_ptrs.begin();
        iter != atoms_ptrs.end(); ++iter)
   {
    kd::KDTree tree1(iter->second);
    for (auto it = std::next(iter); it != atoms_ptrs.end(); ++it)
    {
     for (auto& atom : it->second)
     {
      tree1.search(atom, chdata.CUTNB, pairs_data);
     }
    }
   }
  }
 };
} // psm
```

Листинг 5.3. Реализация оценочной функции с использованием k-d дерева