Геометрия аминокислот и полипептидов

А. О. Иванов, А. С. Мищенко, А. А. Тужилин

предыдущая версия: 3 февраля 2014 г.
текущее обновление: 9 июля 2014 г.

Введение

Задача изучения конформации полипептидов хорошо известна и является чрезвычайно важной. Ее огромное значение связано, а частности, с тем, что изменение конформации входящих в клетку белков, скажем, в следствии мутации, может приводить к серьезным заболеваниям организма. Отметим, что экспериментальное определение последствий той или иной мутации является длительным и дорогим процессом. Поэтому возникает естественное желание — подобные эксперименты смоделировать на компьютере. Однако задача оказывается крайне сложной, и несмотря на усилия огромного научного сообщества, в настоящее время эта задача все еще далека от решения.

Отметим, что обсуждаемая проблема находится на стыке многих наук, а именно, биологии, химии, физики, геометрии, вариационного исчисления, теории вероятностей, вычислительной математики. Возникает насущная необходимость привлечения внимания к этой проблеме ученых самых разных специальностей и создание возможности для заинтересовавшихся быстро войти в курс дела и начать экспериментировать с накопленными данными.

Центральное место в сборе информации о пространственной структуре полипептидов занимает Protein Data Bank (Банк Данных Белков). Первая часть нашей статьи как раз посвящена этой базе. Мы опишем, как можно достаточно быстро выбрать нужную для исследования геометрическую информацию. А затем мы расскажем о целом ряде досадных препятствий, с которыми мы столкнулись при работе с базой.

Когда мы только собирались писать настоящую статью, мы предполагали продемонстрировать, как некоторые геометрические аналогии могут оказаться полезными при изучении конформаций. Но в какой-то момент мы решили проверить данные PDB на “типичность”, чтобы иметь более точное представление о том, с чем мы имеем дело. Результат нашего исследования изложен в следующем параграфе. После ряда “чисток”, мы наконец получили часть базы, которая, хочется верить, описывает “типичное” поведение полипептидов. Мы надеемся, что описанная нами методика тестирования

геометрической базы данных статистико-геометрическими методами окажется полезной для создания более надежных баз такого типа.

В заключительных разделах мы покажем, как можно визуализировать "подвижность" аминокислот, входящих в полипептиды, а также и то, как можно изучать форму полипептидов с помощью двух функций, являющихся аналогами кривизны и кручения пространственных кривых. Отметим, что изучению пространственного строения биологических полимеров посвящено огромное количество работ. Вот некоторые из них [1]— [9], имеющие то или иное отношение к нашему исследованию.

Настоящая статья основа на материалах, опубликованных в [10], [11]. Новые результаты содержатся в основном в разделе 4.

1 Protein Data Bank

> Зона — это ... очень сложная система ... ловушек, что ли ... и все они смертельны!
> ...
> Может даже показаться, что она капризна, но в каждый момент она такова, какой мы ее сами сделали ... своим состоянием.
>
> из фильма А.А.Тарковского "Сталкер"

В данном разделе мы хотим поделиться своими впечатлениями от использования Protein Data Bank (http://www.rcsb.org/pdb/home/home.do) для изучения геометрии аминокислот в составе полипептидов. Собственно говоря, мы хотели извлечь из этой базы информацию о трехмерной структуре полипептидов. Для этого мы решили воспользоваться файлами с расширением pdb: в этих файлах содержатся строки вида

```
ATOM 1 N ASN A 32     65.950  -13.181  61.696     1.00  0.00 N
```

в которых для атома, описанного в 3-ей позиции, указаны его координаты в позициях 7–9. Позиция 4 говорит о том, в какой аминокислоте этот атом содержится, а позиция 2 — номер этой кислоты в рассматриваемом полипептиде. Детали, описывающие структуру формата pdb, можно найти в двухсотстраничном файле

[1] ftp://ftp.wwpdb.org/pub/pdb/doc/format_descriptions/Format_v33_Letter.pdf

Для работы с базой данных мы решили написать свои программы в пакете Mathematica (Wolfram Research Group). Те, кто обладаете быстрым интернетом, могут не скачивать заранее pdb-файлы. Например, чтобы воспользоваться файлом, описывающим полипептид с коротким именем 2LTR, достаточно выполнить команду

```
pdbFile=Import["http://www.rcsb.org/pdb/files/2LTR.pdb", "List"];
```

после этого переменная pdbFile будет представлять собой список строк, содержащихся в файле 2LTR.pdb, и эти строки можно дальше обрабатывать с помощью команд пакета Mathematica, извлекая интересующую информацию. Для работы с другим полипептидом, нужно в приведенной выше команде просто заменить 2LTR на короткое имя интересующего полипептида. Осталось выяснить, где же взять список коротких имен полипептидов?

1.1 Как извлечь нужную нам информацию с PDB?

Вот наш алгоритм:

(1) заходим на сайт http://www.rcsb.org/pdb/home/home.do

(2) в левом верхнем углу нажимаем на кнопку Advanced, расположенную сразу под кнопкой Search

(3) в появившемся разделе Advanced Search Interface

  (a) в выпадающем меню Choose a Query Type выбираем All/Experimantal Type/Molecule Type;
  внизу появляются два поля:
  Experimental Method и Molecule Type

  (b) в Molecule Type выбираем Protein

  (c) справа нажимаем на кнопку Result Count; под этой кнопкой появляется количество выбранных объектов: в нашем случае было написано 89790 PDB Entries (Structures)

  (d) кликаем в эту надпись — внизу получаем список всех выбранных полипептидов, вместе с их короткими именами и многочисленной другой информацией; по умолчанию в каждом квадратике перед коротким именем полипептида стоит галочка — это означает, что данный полипептид выбран.

Конечно, удобней сохранить где-нибудь список всех коротких имен. Для этого кликаем в поле Filter (в строке меню сразу над списком полипептидов) и в выпадающем меню выбираем Download Checked. Открывается страничка Structure Downloaded, в котором есть поле Enter PDB IDs:, где перечисляются все короткие названия выбранных полипептидов (они совпадают с именами соответствующих pdb-файлов). Кликаем в это поле, метим все элементы этого списка (Ctrl-A в Windows), копируем в буфер обмена (Ctrl-C), открываем какой-нибудь текстовый файл (или создаем новый) и копируем содержимое буфера в этот файл (Ctrl-V). Сохраняем файл: теперь в нем содержатся короткие имена всех полипептидов, имеющихся на данный момент в Protein Dada Bank.

Замечание 1.1. Обратите внимание на то, имеется ли пробел после имени в последней строчке полученного файла: если да — уберите его (наличие пробела в нашем случае приводило к ошибкам при выполнении некоторых функций Математики).

Кстати, этим же алгоритмом можно скачать все pdb-файлы с описаниями полипептидов на свой компьютер. Для этого после появления Structure Downloaded, вместо сохранения списка имен, можно

(1) в столбце Download Type: убрать галочку с поля mmCIF Format и поставить галочку в PDB Format;

(2) в столбце Compression Type: переместить точку в поле uncompressed;

(3) внизу нажать Next

(4) теперь Вам нужно, чтобы у Вас был включен JAVA-плагин: это плагин вызовет кнопку Browse, которая позволит выбрать путь для сохранения файлов; выбрав его, нажмите кнопку Start Download. Через некоторое время вы станете обладателем pdb-файлов, описывающих все выбранные полипептиды.

Давайте начнем с визуализации выбранных полипептидов или отдельных их фрагментов.

1.2 Первые шаги самостоятельной визуализации полипептидов

Несложно выделить строки вида "ATOM ..." (см. выше); сгруппировать эти строки по последовательным аминокислотным остатками (в дальнейшем, для краткости, — *аминокислотам*), используя для этого 6-ое поле, где расположен номер аминокислоты в рассматриваемой полипептидной цепи; извлечь из каждой такой строки название атома, его координаты, название аминокислоты, в которую атом входит. Однако, чтобы изобразить структуру связей, эти связи нужно откуда-то взять. В самом pdb-файле связи не представлены. Конечно, структура аминокислот хорошо известна, но в одну и ту же аминокислоту может входить несколько атомов одного и того же типа. Как установить соответствие между последовательными строками "ATOM..." описания одной аминокислоты из pdb-файла и атомами в структурной формуле аминокислоты? В файле [1] такой информации мы не обнаружили. Зато нашли старую версию, а именно,

[2] http://www.wwpdb.org/documentation/PDB_format_1992.pdf

где имеются соответствующие картинки (стр. 27). Впрочем, здесь ничего не сказано про водороды. Но их расположение можно сообразить по обозначениям. Например, если водород записывается как HD, то нужно найти в данной аминокислоте атом или углерода, или азота, или кислорода, или серы, который записывается соответственно через CD, или ND, или OD, или SD. В результате мы построили следующую таблицу связей:

N→CA, CA→C, C→O, CA→CB, C→OXT,
CB→CG, CB→CG1, CB→CG2, CB→OG, CB→OG1, CB→SG,
CG→CD, CG→CD1, CG→CD2, CG→ND1, CG→ND2, CG→OD1, CG→OD2, CG→SD,
CG1→CD1,
CD→CE, CD→NE, CD→OE1, CD→OE2,
CD1→CE1, CD1→NE1,
CD2→CE2, CD2→CE3, CD2→NE2,
CE→NZ, CE→SD,
CE1→CZ, CE1→ND1, CE1→NE2,
CE2→CZ, CE2→CZ2, CE2→NE1,
CE3→CZ3,
CH2→CZ2, CH2→CZ3,
CZ→NE, CZ→NH1, CZ→NH2, CZ→OH,
CA→HA, CA→HA2, CA→HA3,
CB→HB, CB→HB1, CB→HB2, CB→HB3,
CG→HG, CG→HG2, CG→HG3,
CG1→HG11, CG1→HG12, CG1→HG13,
CG2→HG21, CG2→HG22, CG2→HG23,
CD→HD2, CD→HD3, **CD→N**, CD→NE2,
CD1→HD1, CD1→HD11, CD1→HD12, CD1→HD13,
CD2→HD2, CD2→HD21, CD2→HD22, CD2→HD23,
CE→HE1, CE→HE2, CE→HE3,
CE1→HE1,
CE2→HE2,
CE3→HE3,
CH2→HH2,
CZ→HZ,
CZ2→HZ2,
CZ3→HZ3,
N→H, N→H1, N→H2, N→H3,
ND1→HD1,
ND2→HD21, ND2→HD22,
NE→HE,
NE1→HE1,
NE2→HE21, NE2→HE22,
NH1→HH11, NH1→HH12,
NH2→HH21, NH2→HH22,
NZ→HZ1, NZ→HZ2, NZ→HZ3,
OG→HG,
OG1→HG1,
OH→HH,
SG→HG

Используя возможности Mathematica, связанные с рисованием графов, мы научились изображать как отдельные аминокислоты полипептида, так и разные его фрагменты. На рис. 1 приведено несколько примеров изображений аминокислот.

Однако некоторые аминокислоты оказались изображенными неправильно, см. рис. 2.

В чем же дело? Обратите внимание на выделенную связь CD→N в приведенной выше таблице. Оказывается, это она все портит. Мы выяснили, что наличие этой связи — единственная проблема в списке связей. А именно, если все аминокислоты, кроме пролина (рис. 3), изображать с помощью всех оставшихся связей, то ошибок не возникает. Для изображения пролина приходится дополнительно добавлять связь CD→N.

Можно ли решить эту проблему с помощью модификации обозначений атомов аминокислот? Да, это очень просто: достаточно в пролине переименовать CD-атом, скажем, в CF или CW (т.е. дать такое имя, которое не присутствует в названиях атомов других аминокислот). Видимо, разработчикам обозначений это было не нужно.

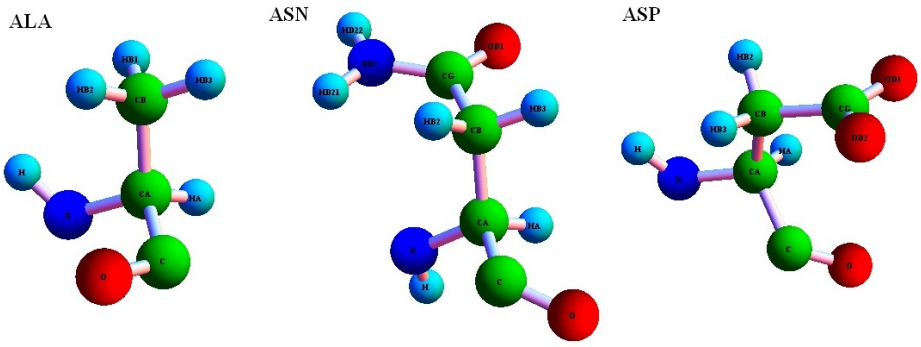

Рис. 1: Пример изображений аминокислот.

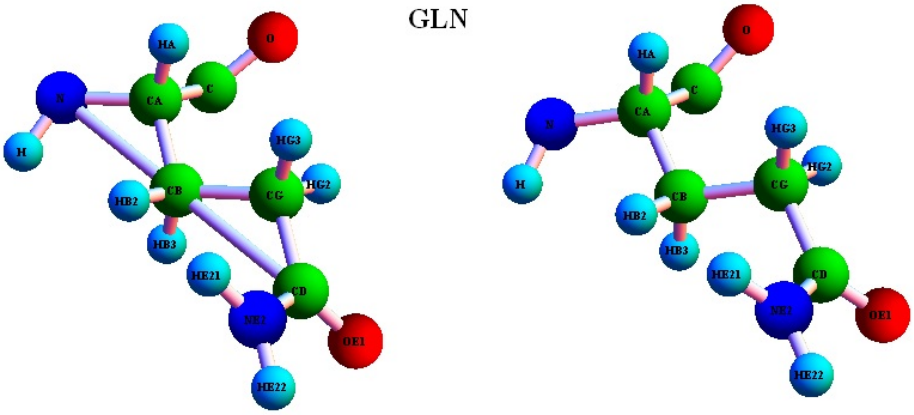

Рис. 2: Слева изображена неправильная структурная формула глутамина. Правильная изображена справа.

Разобравшись со связями, мы решили заняться сравнением геометрии одноименных аминокислот, находящихся в разных местах одного полипептида и, более общо, в разных полипептидах. Но тут мы столкнулись с рядом проблем.

1.3 Ряд трудностей, возникающих при работе с pdb-файлами

(1) Не во всех pdb-файлах присутствуют атомы водорода. Оказалось, что в ряде методов получения трехмерной структуры полипептидов атомы водорода просто не видны, как, скажем, при использовании рентгена.

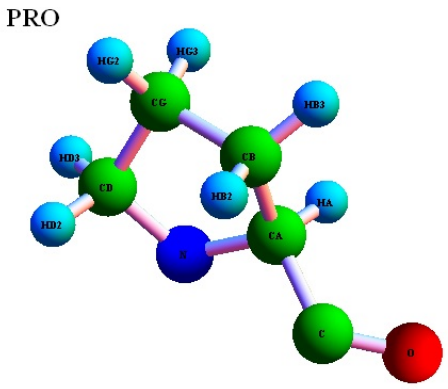

Рис. 3: Пролин.

Мы выяснили, что наиболее подходящим для нас методом является ядерно-магнитный резонанс (NMR). Чтобы выбрать файлы, полученные именно таким методом, мы вернулись на сайт PDB и, в описанном выше алгоритме получения имен всех нужных нам файлов, в поле Experimental Method, которое раньше мы не модифицировали, выбрали NMR. Теперь в Result Count оказалось почти в 10 раз меньше файлов (около 9000).

(2) В следующих файлах не присутствуют строки “ATOM...”:

1R9V.pdb, 1S1O.pdb, 1S4A.pdb, 2KVJ.pdb

(в них имеются только строки “HETATM...”). Эти файлы мы исключили из списка.

(3) Во многих файлах содержится не один вариант полипептида, а несколько. Эти варианты называются *моделями* (models) и наличие нескольких моделей можно отловить по ключевому слову MODEL, с которого начинается строка, стоящая непосредственно перед строками “ATOM...”. Каждая модель начинается с MODEL N, где N — номер модели, и заканчивается строкой “ENDMODEL”. Мы решили выбирать в каждом файле первую модель. Отметим, что непосредственно перед строкой “ENDMODEL...” стоит заключительная строка последовательности строк “ATOM...”, а именно, *терминальная строка* “TER...” (в ней уже нет координат). Тем самым, при обработке pdb-файла мы выкидываем все начало, до первого вхождения строки “ATOM...”, и собираем в список последующие строки до тех пор, пока не появится строка “TER...”.

(4) Описанный только что простой алгоритм выделения первого связного фрагмента натолкнулся на следующее препятствие: оказывается, в некоторых файлах строки между первым вхождением “ATOM...” и первым появлением строки “TER...” могут перемешиваться со строками “HETATM...” (соответствующими *гетерогенным группам*, отличающимся от аминокис-

лотных остатков стандартных 20-и аминокислот). Вот пример из файла 1A13.pdb

```
.......................
ATOM    235 HD22 LEU A  14      -7.905  -4.241   0.457  1.00  0.00          H  ,
ATOM    236 HD23 LEU A  14      -9.150  -4.650   1.637  1.00  0.00          H  ,
HETATM  237  N   NH2 A  15      -4.520  -3.275   5.538  1.00  0.00          N  ,
HETATM  238  HN1 NH2 A  15      -4.461  -2.351   5.201  1.00  0.00          H  ,
HETATM  239  HN2 NH2 A  15      -3.975  -3.556   6.306  1.00  0.00          H  ,
TER     240      NH2 A  15                                                   ,
.......................
```

Мы составили список всех таких файлов (их оказалось 604 штуки) и исключили их из нашего списка.

(5) В некоторых местах аминокислоты может располагаться по несколько копий одного и того же атома (так называемые *альтернативные расположения*). Выяснить, про какое расположение идет речь, можно по 5-ой позиции в строке "ATOM...". Мы решили отбирать лишь те строки "ATOM...", в которых на 5-ом месте стоит буква "A". Кстати, если такие *альтернативные фрагменты* слишком длинные, то они идут отдельным списком, уже после терминальной строки. Мы усовершенствовали наш алгоритм, учтя пятую позицию.

(6) В некоторых файлах (1PNJ.pdb, 2PNI.pdb) несколько последовательных аминокислот метится одним и тем же номером, но различаются полем в следующей позиции, которое обозначается iCode и называется *кодом вставки остатка* (Code for insertion of residues). Вот пример из файла 1PNJ.pdb.

```
.......................
ATOM      1  N   GLY A   1A     18.846  -2.578 -21.181  1.00  0.00          N  ,
ATOM      2  CA  GLY A   1A     19.477  -3.527 -22.142  1.00  0.00          C  ,
ATOM      3  C   GLY A   1A     18.401  -4.379 -22.820  1.00  0.00          C  ,
ATOM      4  O   GLY A   1A     18.321  -4.433 -24.032  1.00  0.00          O  ,
ATOM      5  H1  GLY A   1A     17.815  -2.713 -21.185  1.00  0.00          H  ,
ATOM      6  H2  GLY A   1A     19.216  -2.760 -20.227  1.00  0.00          H  ,
ATOM      7  H3  GLY A   1A     19.069  -1.602 -21.461  1.00  0.00          H  ,
ATOM      8  HA2 GLY A   1A     20.022  -2.968 -22.888  1.00  0.00          H  ,
ATOM      9  HA3 GLY A   1A     20.157  -4.175 -21.609  1.00  0.00          H  ,
ATOM     10  N   SER A   1B     17.604  -5.017 -22.000  1.00  0.00          N  ,
ATOM     11  CA  SER A   1B     16.455  -5.834 -22.505  1.00  0.00          C  ,
ATOM     12  C   SER A   1B     15.466  -5.012 -23.344  1.00  0.00          C  ,
ATOM     13  O   SER A   1B     14.698  -5.571 -24.101  1.00  0.00          O  ,
ATOM     14  CB  SER A   1B     15.737  -6.447 -21.291  1.00  0.00          C  ,
ATOM     15  OG  SER A   1B     16.667  -7.386 -20.765  1.00  0.00          O  ,
ATOM     16  H   SER A   1B     17.763  -4.960 -21.035  1.00  0.00          H  ,
ATOM     17  HA  SER A   1B     16.848  -6.626 -23.126  1.00  0.00          H  ,
ATOM     18  HB2 SER A   1B     15.523  -5.698 -20.541  1.00  0.00          H  ,
ATOM     19  HB3 SER A   1B     14.832  -6.958 -21.575  1.00  0.00          H  ,
ATOM     20  HG  SER A   1B     17.396  -7.490 -21.384  1.00  0.00          H  ,
ATOM     21  N   MET A   1C     15.515  -3.708 -23.185  1.00  0.00          N  ,
ATOM     22  CA  MET A   1C     14.651  -2.769 -23.978  1.00  0.00          C  ,
ATOM     23  C   MET A   1C     13.159  -2.991 -23.648  1.00  0.00          C  ,
ATOM     24  O   MET A   1C     12.312  -3.029 -24.519  1.00  0.00          O  ,
ATOM     25  CB  MET A   1C     14.942  -3.005 -25.496  1.00  0.00          C  ,
ATOM     26  CG  MET A   1C     14.656  -1.721 -26.304  1.00  0.00          C  ,
ATOM     27  SD  MET A   1C     15.658  -0.251 -25.959  1.00  0.00          S  ,
ATOM     28  CE  MET A   1C     17.288  -0.860 -26.468  1.00  0.00          C  ,
ATOM     29  H   MET A   1C     16.138  -3.336 -22.527  1.00  0.00          H  ,
ATOM     30  HA  MET A   1C     14.912  -1.756 -23.706  1.00  0.00          H  ,
ATOM     31  HB2 MET A   1C     15.976  -3.283 -25.625  1.00  0.00          H  ,
ATOM     32  HB3 MET A   1C     14.325  -3.806 -25.874  1.00  0.00          H  ,
ATOM     33  HG2 MET A   1C     14.777  -1.962 -27.348  1.00  0.00          H  ,
ATOM     34  HG3 MET A   1C     13.620  -1.444 -26.162  1.00  0.00          H  ,
```

```
ATOM     35  HE1 MET A   1C     17.204  -1.841 -26.912  1.00  0.00           H  ,
ATOM     36  HE2 MET A   1C     17.720  -0.181 -27.189  1.00  0.00           H  ,
ATOM     37  HE3 MET A   1C     17.934  -0.915 -25.603  1.00  0.00           H  ,
.........................
```

Игнорирование этого привело нас к “аминокислоте” с атомным составом

N, CA, C, O, H1, H2, H3, HA2, HA3, N, CA, C, O, CB, OG, H, HA, HB2, HB3, HG, N, CA, C, O, CB, CG, SD, CE, H, HA, HB2, HB3, HG2, HG3, HE1, HE2, HE3.

Мы расширили признак сборки аминокислот из строк “ATOM...” на позицию iCode, в результате чего эта “аминокислота” развалилась на три: GLY, SER, MET (глицин, серин, метионин), см. рис. 4.

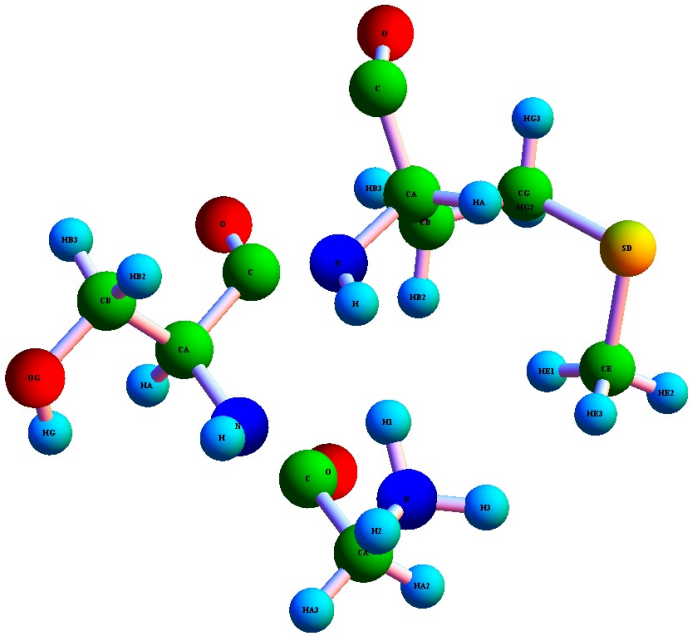

Рис. 4: Три различные аминокислоты, по ошибке распознанные как одна.

(7) Аминокислоты, имеющие одни и те же имена, могут иметь разный атомный состав. Естественно, концевые аминокислоты должны отличаться от неконцевых: на N-конце присутствует пара “лишних” водородов, а на C-конце — “лишний” кислород. Тем самым, мы решили ограничиться *внутренними* (неконцевыми) аминокислотами.

Однако оказалось, что и внутренние одноименные аминокислоты могут иметь разное атомное строение! Например, у цистеина к атому серы иногда крепится атом водорода, а иногда — нет, см. рис. 5.

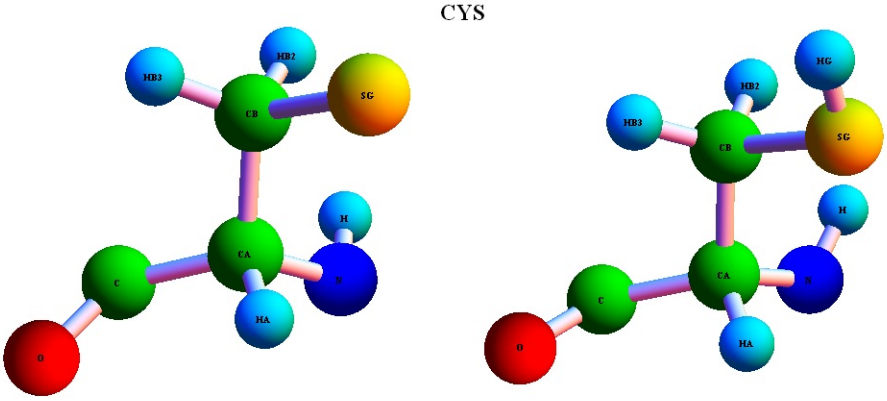

Рис. 5: Различные атомные составы внутреннего цистеина.

Мы прошли по всем файлам, и для каждой аминокислоты определили все возможные *атомные составы* аминокислот, содержащихся в выбранных полипептидах. Параллельно мы вычислили, сколько раз встречается каждая из полученных последовательностей атомов. Вот пример для глицина GLY:

```
{55705,{N,CA,C,O,H,HA2,HA3}},{1576,{N,CA,C,O,H1,HA2,HA3}},
{870,{N,CA,C,O,H1,H2,H3,HA2,HA3}},{399,{N,CA,C,O,OXT,H,HA2,HA3}},
{390,{N,CA,C,O}},{384,{N,CA,C,O,H}},{47,{N,CA,C,O,H,HA2}},
{37,{CA}},{37,{N,CA,C,O,HA2,HA3}},{26,{N,CA,C,O,H2,HA2,HA3}},
{15,{N,CA,C,O,H1,H2,HA2,HA3}},{15,{N,CA,C,O,HA2,HA3,H1,H2,H3}},
{13,{N,CA,C,O,H,HA}},{13,{N,CA,C,O,HA2, HN,HA1}}},{9,{N,CA,C}},
{8,{N,CA,C,O,H1,H2,H3}},{7,{C,CA,H,HA2,HA3,N,O}},
{5,{N,CA,C,O,OXT,H}},{5,{N,CA,C,H,HA2,HA3}},
{4,{N,CA,C,O,H,HA3,HA2}},{2,{N,CA,C,O,H1,H2,H3,HA3}},
{2,{N,CA,C,O,HA2,HA3,H1,H2}},{2,{N,CA,C,O,OXT}},{2,{CA,C}},
{1,{N,CA,C,O,OXT,H,HA2,HA3,HXT}},{1,{N,CA,C,O,H,HA2,HA3,H2,H3}},
{1,{N,CA,C,O,H2,H,H3,HA2,HA3}},{1,{N,CA,C,O,H2,H,HA2,HA3}},{1,{N}},
{1,{N,CA,C,OXT,H,HA2,HA3}},{1,{N,CA,C,O,HA2}},{1,{N,CA,C,O,H,HA3}}.
```

Затем мы выбрали наиболее часто встречающиеся последовательности и назвали их *стандартными*. Вот полученный список стандартных атомных составов аминокислот, вместе с количествами их появлений:

```
(GLY:55705) N,CA,C,O,H,HA2,HA3;
(ALA:50673) N,CA,C,O,CB,H,HA,HB1,HB2,HB3;
(SER:52599) N,CA,C,O,CB,OG,H,HA,HB2,HB3,HG;
(CYS:11784) N,CA,C,O,CB,SG,H,HA,HB2,HB3;
(PRO:32529) N,CA,C,O,CB,CG,CD,HA,HB2,HB3,HG2,HG3,HD2,HD3;
(VAL:48533) N,CA,C,O,CB,CG1,CG2,H,HA,HB,HG11,HG12,HG13,HG21,HG22,HG23;
(THR:39499) N,CA,C,O,CB,OG1,CG2,H,HA,HB,HG1,HG21,HG22,HG23;
```

```
(ILE:37003) N,CA,C,O,CB,CG1,CG2,CD1,
                      H,HA,HB,HG12,HG13,HG21,HG22,HG23,HD11,HD12,HD13;
(LEU:60276) N,CA,C,O,CB,CG,CD1,CD2,
                        H,HA,HB2,HB3,HG,HD11,HD12,HD13,HD21,HD22,HD23;
(ASP:40010) N,CA,C,O,CB,CG,OD1,OD2,H,HA,HB2,HB3;
(ASN:29276) N,CA,C,O,CB,CG,OD1,ND2,H,HA,HB2,HB3,HD21,HD22;
(GLU:53111) N,CA,C,O,CB,CG,CD,OE1,OE2,H,HA,HB2,HB3,HG2,HG3;
(GLN:29973) N,CA,C,O,CB,CG,CD,OE1,NE2,H,HA,HB2,HB3,HG2,HG3,HE21,HE22;
(MET:14570) N,CA,C,O,CB,CG,SD,CE,H,HA,HB2,HB3,HG2,HG3,HE1,HE2,HE3;
(LYS:50343) N,CA,C,O,CB,CG,CD,CE,NZ,
                      H,HA,HB2,HB3,HG2,HG3,HD2,HD3,HE2,HE3,HZ1,HZ2,HZ3;
(ARG:35509) N,CA,C,O,CB,CG,CD,NE,CZ,NH1,NH2,
                    H,HA,HB2,HB3,HG2,HG3,HD2,HD3,HE,HH11,HH12,HH21,HH22;
(HIS: 8040) N,CA,C,O,CB,CG,ND1,CD2,CE1,NE2,H,HA,HB2,HB3,HD1,HD2,HE1;
(PHE:26312) N,CA,C,O,CB,CG,CD1,CD2,CE1,CE2,CZ,H,HA,HB2,HB3,
                                                    HD1,HD2,HE1,HE2,HZ;
(TYR:22457) N,CA,C,O,CB,CG,CD1,CD2,CE1,CE2,CZ,
                                    OH,H,HA,HB2,HB3,HD1,HD2,HE1,HE2,HH;
(TRP: 8998) N,CA,C,O,CB,CG,CD1,CD2,NE1,CE2,CE3,CZ2,CZ3,CH2,
                                  H,HA,HB2,HB3,HD1,HE1,HE3,HZ2,HZ3,HH2.
```

Замечание 1.2. Отметим, что все выбранные типы атомных составов отвечают внутренним аминокислотам, что вполне естественно, так как концевые должны встречаться намного реже.

Отметим также, что "популярности" некоторых типов аминокислот бывают вполне сравнимы. Так, количества появлений типов изображенных на рис. 5 внутренних цистеинов отличаются в нашем эксперименте менее чем в 2 раза (тип, расположенный слева, встречается чаще). Приведем, для сравнения, их последовательности и число встреч:

```
11784, {N,CA,C,O,CB,SG,H,HA,HB2,HB3},
 6254, {N,CA,C,O,CB,SG,H,HA,HB2,HB3,HG}.
```

Замечание 1.3. Легко видеть, что атом H (который "крепится" к азоту из пептидного остова) присутствует не во всех отобранных атомных составах аминокислот. А именно, такой аминокислотой является пролин PRO, рис. 3. И это вполне естественно, в силу наличия связи CD→N, которую мы обсуждали выше.

Тем не менее, мы решили более внимательно посмотреть на отброшенные атомные составы пролина: может, в некоторых из них все-таки присутствует водород? И мы нашли такие составы:

```
10, {N,CA,C,O,CB,CG,CD,H,H3,HA,HB2,HB3,HG2,HG3,HD2,HD3},
 1, {N,CA,C,O,H,HA,HB2},
 1, {N,CA,C,O,CB,CG,CD,H3,H,HA,HB2,HB3,HG2,HG3,HD2,HD3},
 1, {N,CA,C,O,CB,CG,CD,H,HA,HB2,HB3,HG2,HG3,HD2,HD3},
 1, {N,CA,C,O,CB,CG,CD,HA,HB2,HB3,HG2,HG3,HD2,H}.
```

Здесь число в каждой записи указывает на количество вхождений аминокислот идущего далее атомного состава. Заметим, что таких вхождение

совсем мало, а именно 14, против 32529 самых популярных.

В заключение этого этапа, мы прошли по всем pdb-файлам и выделили те, в которых все внутренние аминокислоты, встречающиеся в первой модели, — стандартные. Наша база сократилась примерно в 3 раза (теперь она содержит 2892 файла). После этого мы преобразовали каждый файл в файл "короткого формата", собрав там именно ту информацию, которая нам нужна для текущих исследований. Теперь в выбранных файлах у нас лежат списки вида

$$
\begin{Bmatrix}
\\
\{\text{ИмяПервойВнутреннейАминокислоты},\\
\quad \{\text{КоординатыПервогоАтома, КоординатыВторогоАтома, } \ldots\}\},\\
\ldots\\
\{\text{ИмяПоследнейВнутреннейАминокислоты},\\
\quad \{\text{КоординатыПервогоАтома, КоординатыВторогоАтома, } \ldots\}\},\\
\end{Bmatrix}
$$

Таким образом, если начальная база полипептидов, полученных методом NMR, занимала больше 18 гигабайт, то нынешняя занимает около 300 мегабайт. Конечно же, функции, необходимые для наших дальнейших исследований, с файлами новой базы работают на порядок быстрее.

2 Метрический анализ PDB

Получив наконец одинаковый атомный состав внутренних аминокислот в изучаемых полипептидах, мы вернулись к задаче сравнения геометрии одноименных аминокислот. Для этого мы сначала решили понять, сколь сильно могут варьироваться длины ковалентных связей, и как могут меняться углы между связями.

2.1 Оценка разброса длин ковалентных связей в аминокислотах

Для оценки отклонений длин ковалентных связей в данном наборе полипептидов мы решили поступить так:

(1) выбрать из всех полипептидов внутренние аминокислоты данного типа, например, все глицины;

(2) вычислить средние значения длины каждой ковалентной связи по всем выбранным аминокислотам;

(3) для каждого типа аминокислот (например, для глицинов) и каждой ковалентной связи в аминокислотах этого типа (например для связи N-H) вычислить максимальное, по всем аминокислотам этого типа,

относительное отклонение от среднего значения (для удобства, это отклонение мы измеряли в процентах).

Для небольших фрагментов нашей базы эти отклонения были достаточно маленькие (несколько процентов). Однако, когда мы запустили программу на всю базу, то обнаружили огромные отклонения. Самое большое из них было у аланина (705.819% для связи N-H):

$$\frac{454.736}{\mathrm{N-CA}}, \frac{362.668}{\mathrm{CA-C}}, \frac{77.8317}{\mathrm{C-O}}, \frac{244.119}{\mathrm{CA-CB}}, \frac{328.258}{\mathrm{CA-HA}}, \frac{105.323}{\mathrm{CB-HB1}}, \frac{185.751}{\mathrm{CB-HB2}}, \frac{152.231}{\mathrm{CB-HB3}}, \frac{705.819}{\mathrm{N-H}}$$

(числитель каждой дроби — максимальное процентное отклонение ковалентной связи, указанной в знаменателе, от среднего значения).

Мы "отловили" файл, который привел к таким отклонениям. Им оказался 2PDE.pdb. Мы решили выяснить, как выглядят аминокислоты из этого файла. Результаты превзошли наши ожидания, см. рис. 6.

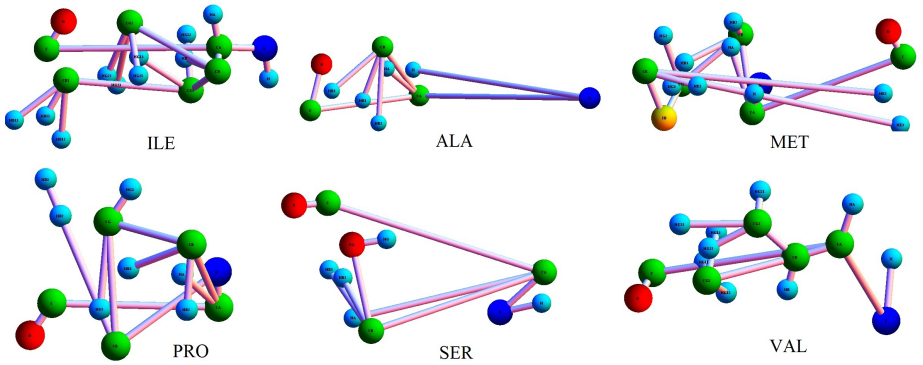

Рис. 6: Некоторые "аминокислоты" из файла 2PDE.pdb.

Отметим, что последние 8 аминокислот в этом файле выглядят стандартно.

Следующий "рекордсмен отклонений по аланину" — файл 2HQ3.pdb, см. рис. 7.

Впрочем, после выкидывания файла 2PDE.pdb, отклонения по глицину стали больше, чем по аланину. "Виновником" оказался файл 2I2J.pdb, см. рис. 8.

Мы отобрали все файлы, в которых отклонение не только по глицину, но и по остальным аминокислотам, больше 15% — таких файлов оказалось не так много, а именно, 30 штук. Вот список их имен:

1AC0, 1BBA, 1COD, 1DOQ, 1GXX, 1J4M, 1K76, 1KWD, 1KWE, 1L0M, 1L8Z, 1MZI, 1OT4, 1PGY, 1XEE, 1Y7J, 2CYK, 2EVZ, 2HQ3, 2I2J, 2JMM, 2K1W, 2KZG, 2L9V, 2NR1, 2PDE, 3ZGK, 3ZOB, 3ZPK, 4BXU.

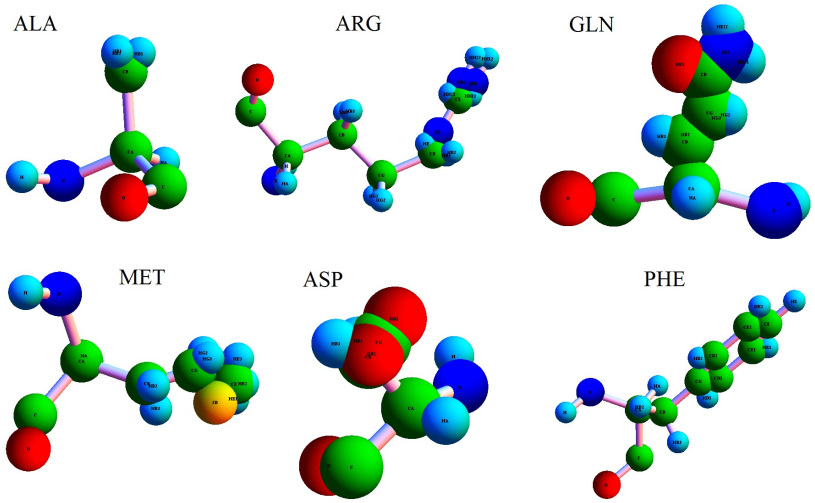

Рис. 7: Некоторые “аминокислоты” из файла 2HQ3.pdb.

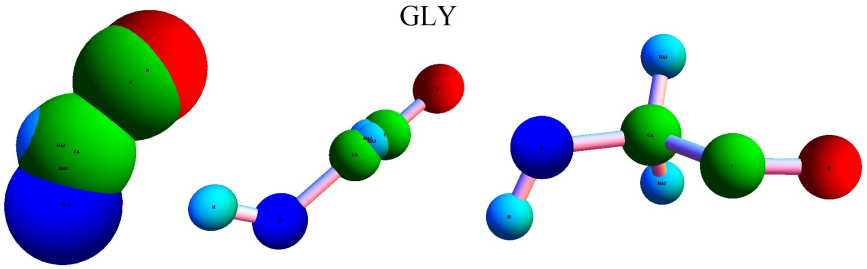

Рис. 8: Слева: два “глицина” из файла 2I2J.pdb. Справа — “стандартный” глицин.

Выбросив из базы эти файлы, мы получили максимальное отклонение равным 14.6% (у валина):

$$\frac{8.89612}{\mathrm{N}\to\mathrm{CA}},\ \frac{6.43247}{\mathrm{CA}\to\mathrm{C}},\ \frac{4.27047}{\mathrm{C}\to\mathrm{O}},\ \frac{8.92551}{\mathrm{CA}\to\mathrm{CB}},\ \frac{5.34567}{\mathrm{CB}\to\mathrm{CG1}},\ \frac{7.08376}{\mathrm{CB}\to\mathrm{CG2}},\ \frac{14.6298}{\mathrm{CA}\to\mathrm{HA}},$$
$$\frac{9.57366}{\mathrm{CB}\to\mathrm{HB}},\ \frac{7.72194}{\mathrm{CG1}\to\mathrm{HG11}},\ \frac{7.7138}{\mathrm{CG1}\to\mathrm{HG12}},\ \frac{7.71296}{\mathrm{CG1}\to\mathrm{HG13}},\ \frac{7.70222}{\mathrm{CG2}\to\mathrm{HG21}},$$
$$\frac{7.69787}{\mathrm{CG2}\to\mathrm{HG22}},\ \frac{7.7378}{\mathrm{CG2}\to\mathrm{HG23}},\ \frac{10.1827}{\mathrm{N}\to\mathrm{H}}.$$

Интересно, как распределено количество файлов с теми или иными отклонениями? Чтобы это выяснить, мы разбили отрезок от -15 до +15 на ма-

лые отрезки длины 0.1 и вычислили, сколько файлов имеют максимальные отклонения, попадающие на тот или иной из малых отрезков (в Mathematica это удобно сделать с помощью функции BinCounts). Результаты мы изобразили на графике (по абсциссе отложены номера малых отрезков, по ординате — количество файлов с максимальными отклонениями, попавшими на соответствующий отрезок). Графики оказались очень похожи друг на друга. На рис. 9 приведен пример для глицина.

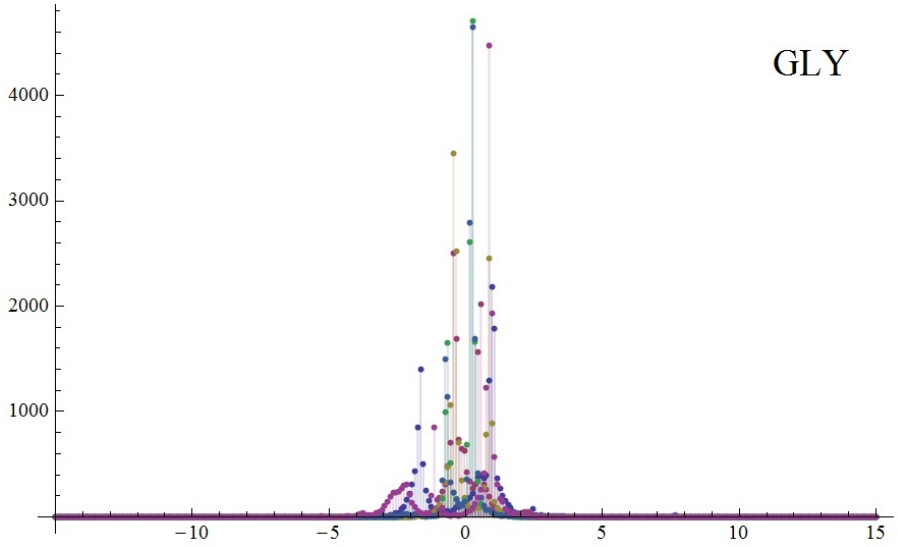

Рис. 9: Распределение количеств файлов с теми или иными максимальными отклонениями от средних значений длин ковалентных связей в глицине.

Непосредственное рассмотрение всех 20 графиков позволило сделать вывод, что для основной массы отклонения не превышают 5%. Мы решили выяснить, сколько файлов для каждой аминокислоты не укладывается в 5%-ый интервал. Ниже приведен ответ.

GLY : 28; ALA : 26; SER : 47; CYS : 10; PRO : 157; VAL : 37; THR : 43; ILE : 31; LEU : 38; ASP : 29; ASN : 33; GLU : 41; GLN : 30; MET : 15; LYS : 53; ARG : 48; HIS : 122; PHE : 26; TYR : 42; TRP : 183.

Как видно, аминокислоты разбиваются на две группы: 17 штук, у которых "большие" отклонения присутствуют в 10–53 полипептидах, и 3 штуки, у которых отклонения имеются в 122–183 полипептидах (HIS, Pro, TRP). На рис. 10 приведен пример для триптофана.

Отметим, что в приведенной выше таблице количеств файлов с "большими отклонениями" общее количество, а именно, 1039, больше, чем реальное число файлов (некоторые файлы относятся одновременно к нескольким типам аминокислот). Это реальное число файлов равно 442. Так как

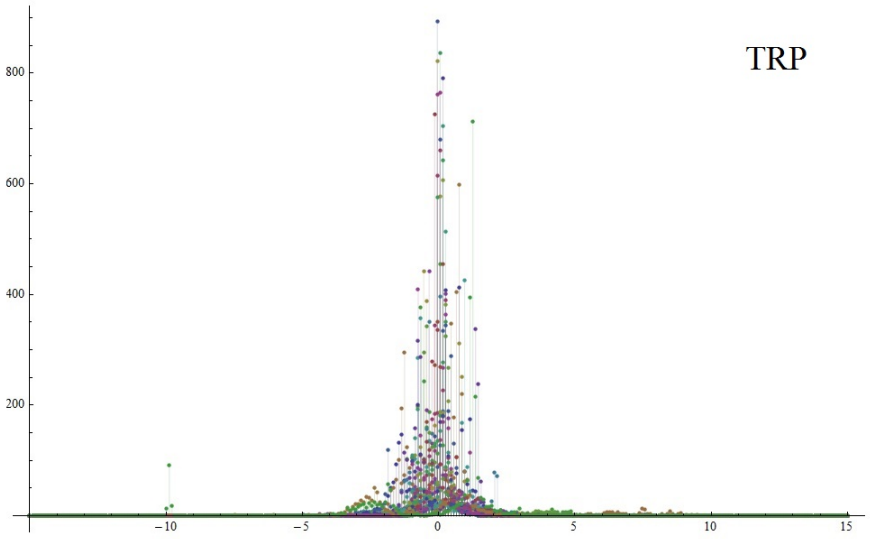

Рис. 10: Распределение количеств файлов с теми или иными максимальными отклонениями от средних значений длин ковалентных связей в триптофане.

к настоящему моменту в нашей базе имелось 2862 файлов, число файлов с "большими отклонениями" оказалось достаточно значимым. В следующей таблице приводятся количества файлов с отклонениями большими, чем n%, для $n = 6, \ldots, 12$:

$$6\% : 371;\ 7\% : 279;\ 8\% : 162;\ 9\% : 128;\ 10\% : 20;\ 11\% : 8;\ 12\% : 6.$$

Так как мы для начала интересуемся типичными метрическими свойствами полипептидов, мы решили поместить в особый список 20 файлов, соответствующих 10%, и исключить их из нашего основного списка. Тем самым, мы удалили файлы с именами

1ALE, 1ALF, 1C7V, 1E0Q, 1E2B, 1IFY, 1LVQ, 1MEQ, 1ODP, 1ODR, 1OEF, 1OEG, 1OPP, 1PFT, 1SOL, 1SVR, 1TXA, 1ZRP, 2BAU, 2KOX

и получили базу данных из 2842 файлов, в которой отклонение длин ковалентных связей в аминокислотах не превышает 10%.

Следующий шаг — изучить отклонение от среднего расстояний между альфа-углеродами последовательных внутренних аминокислот в отобранных выше полипептидах. *В дальнейшем мы будем обозначать альфа-углероды в двух последовательных аминокислотах через* CA-CA.

2.2 Оценка разброса длин расстояний между последовательными альфа-углеродами (начало)

Для каждого полипептида мы вычислили максимальное относительное отклонение расстояний CA-CA и вывели график этих отклонений, см. рис. 11. По абсциссе отложен номер полипептида, а по ординате — максимальное отклонение.

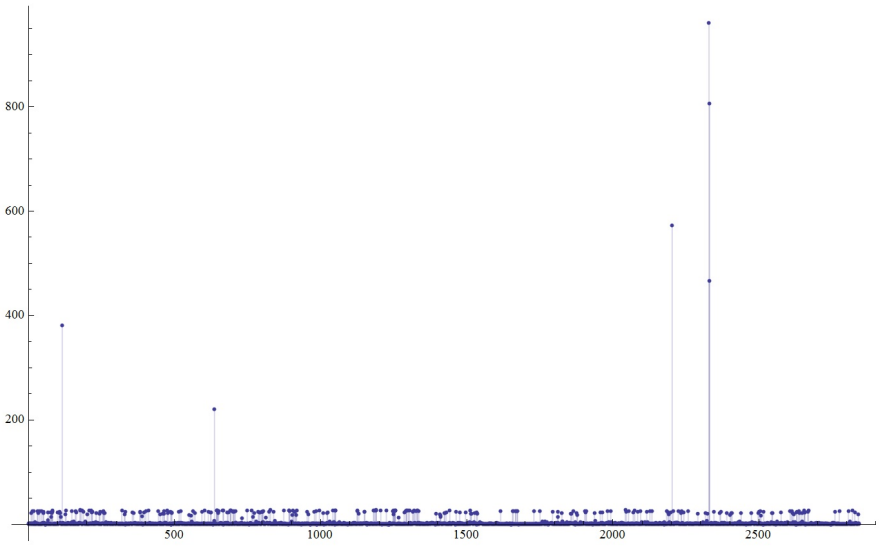

Рис. 11: Максимальные относительные отклонения от среднего расстояний между последовательными α-углеродами в отобранных выше полипептидах.

Обратите внимание на небольшое количество "выбросов". Мы нашли соответствующие полипептиды. Вот их список: 1BH7, 1JJS, 2KSE, 2L6F, 2L6G, 2L6H. Причина оказалась простой: некоторые аминокислоты не были распознаны в эксперименте. Впрочем, в этих файлах имеются соответствующие указания. Во-первых, в комментариях можно найти надпись следующего типа (для примера, рассмотрим 1BH7.pdb):

```
REMARK 465 MISSING RESIDUES
REMARK 465 THE FOLLOWING RESIDUES WERE NOT LOCATED IN THE
REMARK 465 EXPERIMENT. (RES=RESIDUE NAME; C=CHAIN IDENTIFIER;
REMARK 465 SSSEQ=SEQUENCE NUMBER; I=INSERTION CODE.)
REMARK 465     RES C SSSEQI
REMARK 465     LEU A    17
REMARK 465     VAL A    28
```

Во-вторых, в списке "ATOM..." 6-ая позиция, отвечающая за номер ами-

нокислоты, также демонстрирует пропущенные аминокислоты. Скажем, в том же файле после номера 16 сразу следует номер 18.

Выбросив из нашей базы эти 6 файлов, мы получили максимальные относительные отклонения не более 30%. На рис. 12 показано, как распределены количества файлов (ордината) с теми или иными максимальными отклонениями (абсцисса): 1 по абсциссе соответствует изменению величины отклонения на 1%.

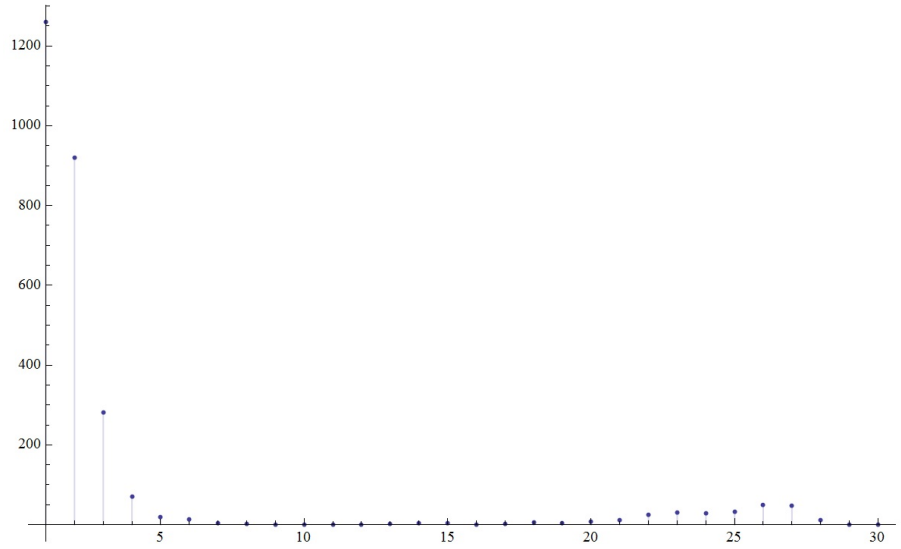

Рис. 12: Количества файлов (ордината) с теми или иными максимальными отклонениями (абсцисса) в CA–CA связях.

А вот соответствующие количества:

$$1260, 920, 282, 70, 20, 13, 4, 2, 1, 0, 0, 0, 2, 3, 3, 1, 2, 5, 3, 8, 12, 24, 30, 28, 33, 49, 48, 12, 1, 0$$

Замечание 2.1. Интересно, что хотя основная масса файлов имеет отклонения не более 5–6%, тем не менее, имеется второе “скопление” в окрестности 25%. Ниже мы объясним, в чем дело.

Конечно, одной из очевидных причин наличия “скопления” мог бы оказаться разброс длин пептидных связей C-N.

2.3 Оценка разброса длин пептидных связей

Мы изучили отклонения в пептидных связях C-N (между последовательными аминокислотами). Эти отклонения оказались не превосходящими

7.8%. На рис. 13 приведено распределение количеств полипептидов с максимальными отклонениями, отложенными по оси абсцисс (шаг по абсциссе равен отклонению на 0.2%).

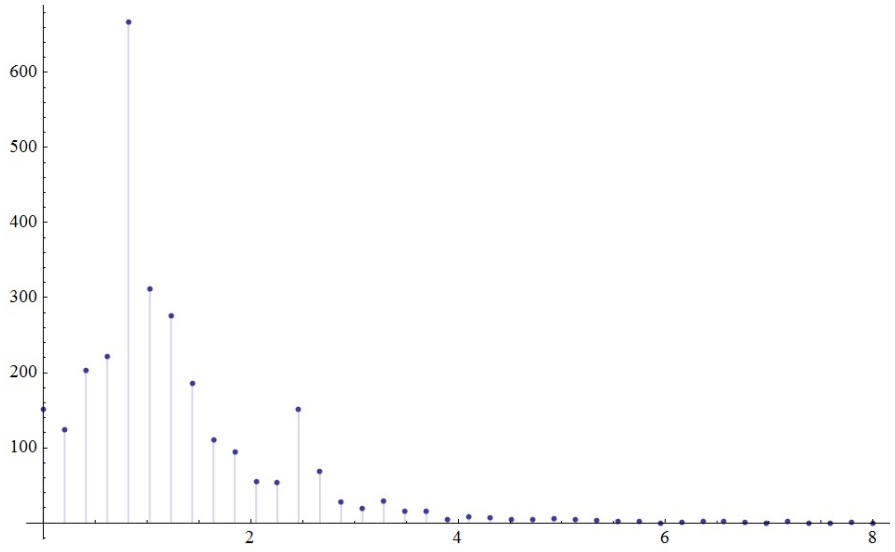

Рис. 13: Количества файлов (ордината) с теми или иными отклонениями (абсцисса) в C–N связях.

Таким образом, "скопления" в разбросе расстояний CA-CA обусловлены не этим.

2.4 Закон плоскости

В биохимии традиционно выделяются следующие шестерки атомов пептидного остова: атомы CA, C, O для $(i-1)$-ой аминокислоты и CA, N, H для i-ой аминокислоты. Такие шестерки называются *пептидными группами*. Считается, что каждая пептидная группа лежит в одной плоскости. Это утверждение будем называть *законом плоскости*. Естественно, нарушение этого закона также влияет на величину разброса расстояний CA-CA.

Проверим этот закон, взяв случайный полипептид, скажем, 1CKR, и в нем — случайную пару последовательных аминокислот, например, с номерами 91 (VAL) и 92 (THR), см. рис. 14.

Отчетливо видно, что в этом случае пептидная группа образует почти плоскую конфигурацию, т.е. закон плоскости имеет место.

Нам повезло, что второй выбранной аминокислотой не оказался пролин PRO, ведь у пролина нет атома H. Давайте посмотрим, что происходит, если вторая аминокислота — пролин, рис. 15.

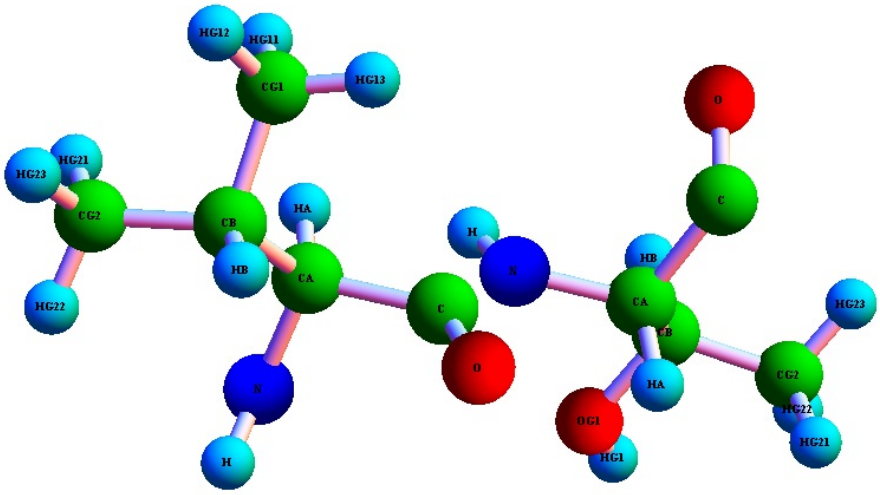

Рис. 14: Иллюстрация утверждения о том, что атомы CA, C, O для $(i-1)$-ой аминокислоты и CA, N, H для i-ой аминокислоты лежат в одной плоскости.

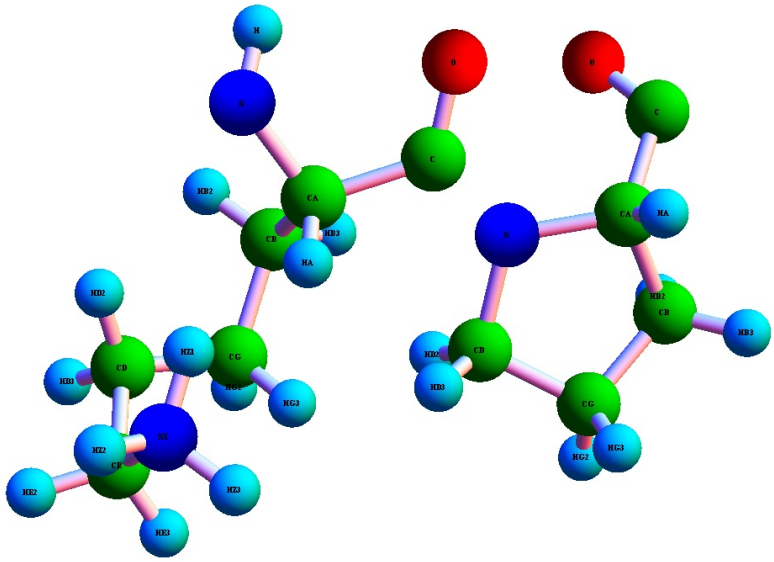

Рис. 15: Последовательные аминокислоты, вторая — пролин.

Из рисунка видно, как обобщить закон плоскости и на этот случай: нуж-

но взять вместо атома H атом CD. *В дальнейшем, говоря о законе плоскости, мы будем иметь в виду именно это обобщение.*

Чтобы проверить действие закона плоскости во всех остальных случаях, мы написали программу, которая для каждой пары последовательных аминокислот строит выпуклую оболочку атомов пептидной группы и вычисляет объем этой оболочки. На первых попавшихся полипептидах максимальное значение объема было равно примерно 0.5. Но на всей совокупности полипептидов максимальное значение оказалось равным 4.74761. Мы нашли соответствующие полипептид и пару его последовательных аминокислот. Ими оказались полипептид 1PBA и его 38 (LYS) и 39 (PRO) аминокислоты. И вот что мы увидели, рис. 16.

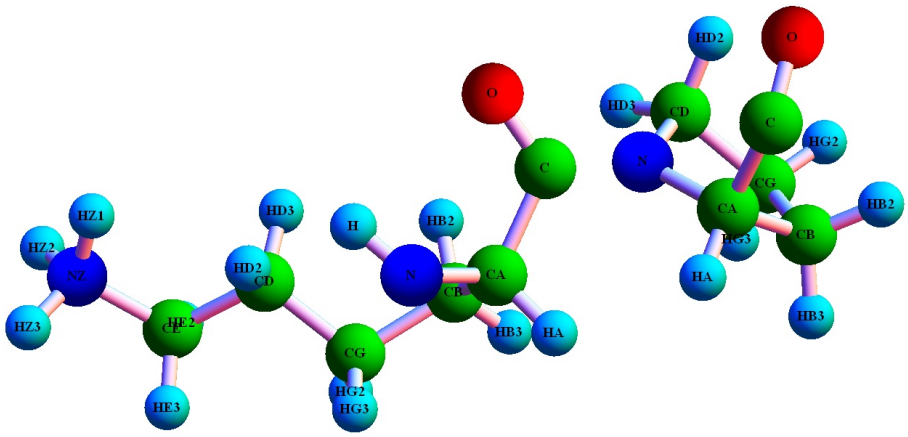

Рис. 16: “Контрпример” к утверждению о том, что атомы CA, C, O для $(i-1)$-ой аминокислоты и N, H, CA для i-ой аминокислоты лежат в одной плоскости (“особая” конфигурация).

Приведенный на рис. 16 пример является особым в том смысле, что вторая аминокислота в нем — пролин (напомним, именно такими конфигурациями мы расширили закон плоскости). Отметим, что, вообще говоря, для таких особых конфигураций значения их объемов могут измерять степень неплоскости конфигурации пептидной группы по-другому по отношению к объемам для неособых конфигураций. И это действительно имеет место: объемы для особых конфигураций с аналогичными отклонениями, что и у неособых, меньше. И этот факт хорошо виден на следующем примере неособой конфигурации, на которой достигается максимум объема (4.4387) в классе неособых конфигураций, рис. 17.

Чтобы оценить, как связаны объемы со степенью плоскости, мы рассмотрели выпуклые оболочки пептидных групп с разными объемами, выбрали некоторые из них и постарались сориентировать так, чтобы проекции на фронтальную плоскость имели минимальную высоту (это высота и

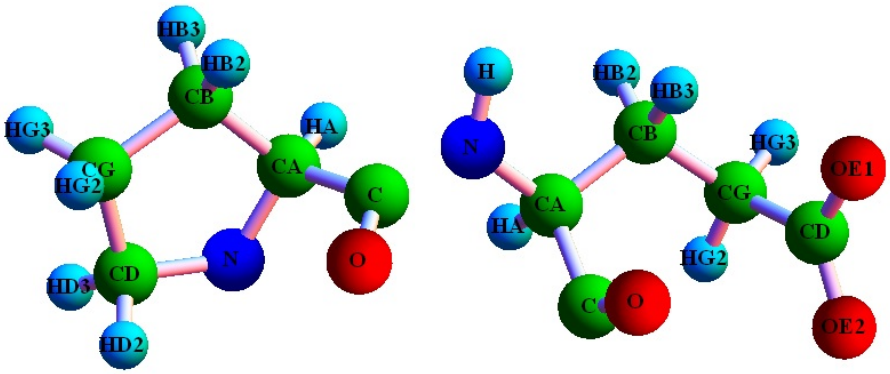

Рис. 17: “Контрпример” к утверждению о том, что атомы CA, C, O для $(i-1)$-ой аминокислоты и CA, N, H для i-ой аминокислоты лежат в одной плоскости (“неособая” конфигурация).

оценивает то, сколь конфигурация плоская: чем меньше высота, тем более плоская конфигурация). На рис. 18 показаны примеры (для неособых конфигураций) и подписаны объемы. Видно, что даже для объема порядка 1 конфигурация совсем плоской не выглядит.

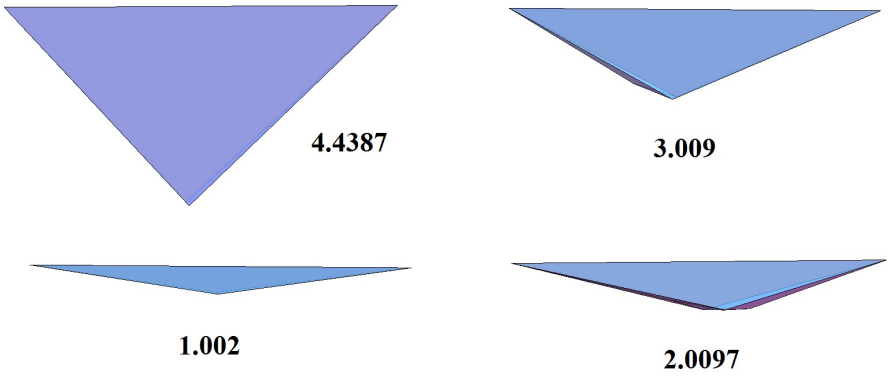

Рис. 18: Проекции выпуклых оболочек на фронтальную плоскость, имеющие наименьшие высоты.

На рис. 19 показано, как выглядят соответствующие пары аминокислот для объемов ~ 1 и ~ 2 из рис. 18.

Мы решили посчитать распределение количеств полипептидов, в которых максимальные отклонения объемов лежат в том или ином интервале.

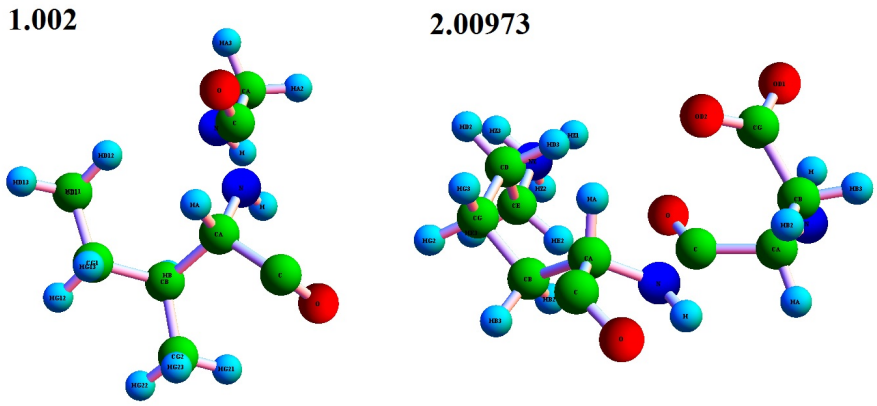

Рис. 19: Пары аминокислот, соответствующие объемам ~ 1 и ~ 2 из рис. 18.

Для этого мы разбили отрезок $[0, 4.8]$ на малые отрезки длины 0.05, и для каждого из полученных малых отрезков посчитали, у скольких полипептидов максимальные объемы выпуклых оболочек рассматриваемых пептидных групп лежат на этом отрезке. На рис. 20 показан соответствующий график.

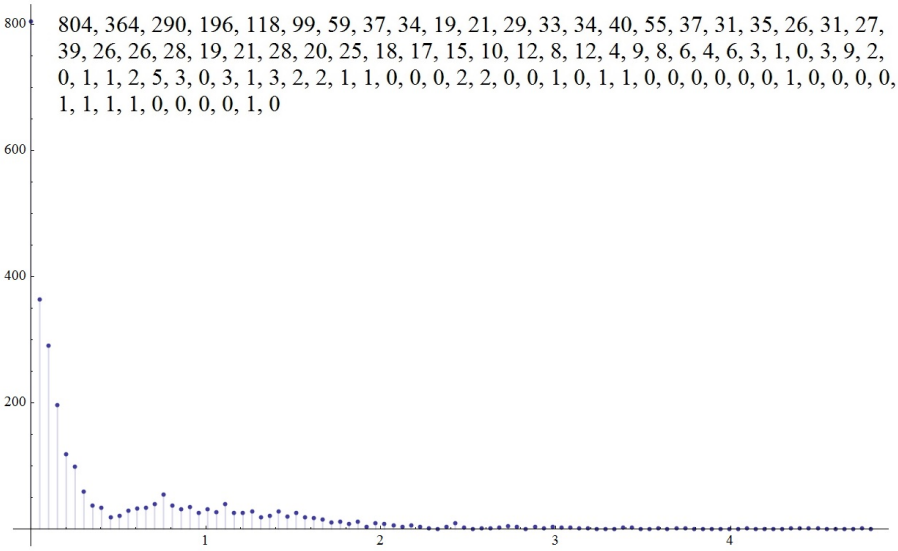

Рис. 20: Распределение количеств полипептидов (ордината) по значениям объемов выпуклых оболочек пептидных групп (абсцисса).

Замечание 2.2. Несложно вычислить, что максимальный объем больше 1 имеют 475 полипептидов (из 2836), т.е. 16.7%. Как мы уже отмечали выше, при таком объеме хорошо заметно, что конфигурация соответствующей шестерки неплоская. Таким образом, в каждом из 16.7% полипептидов имеется аминокислота, в которой хорошо заметно отклонение от закона плоскости.

С другой стороны, мы рассмотрели все возможные последовательные пары аминокислот в имеющихся полипептидах. Их оказалось 190185 штук. Считая, что закономерность хорошая, если она выполняется для 95% случаев, мы получили следующий результат: количество пар последовательных аминокислот, в которых выпуклые оболочки шестерок имеют объем меньше 0.575, составляет 95% от общего количества пар последовательных аминокислот. На рис. 21 показан пример для максимального объема 0.575008.

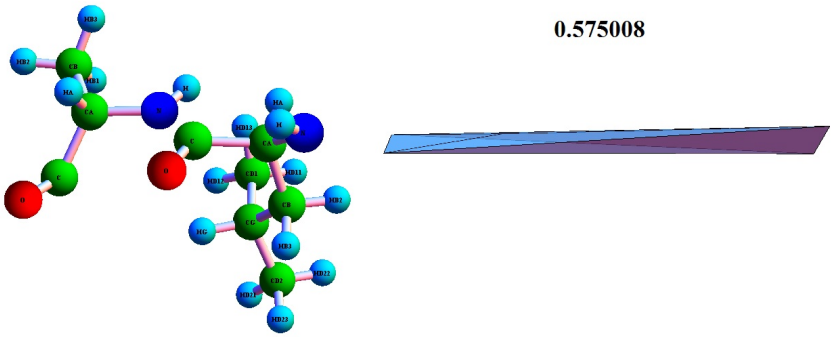

Рис. 21: Пара аминокислот, реализующая максимальный объем 0.575008 для 95% пар последовательных аминокислот.

Заметим, что здесь закон плоскости выполняется с хорошей точностью. Таким образом, можно считать, что этот закон прошел наш “тест на выживание”. Впрочем, ряд полипептидов все-таки стоит исключить из рассмотрения. Если исходить из желания оставить около 95% полипептидов, то нужно исключить все те, у которых объемы превышают 1.67. Тогда останется 95.1%, а исключено будет 138 файлов.

2.5 Оценка разброса длин расстояний между последовательными альфа-углеродами (окончание)

Мы решили сделать следующий “радикальный” эксперимент: исключить все файлы, в которых максимальные объемы превышают 0.5 (в оставшихся файлах закон плоскости выполняется с большой точностью) и посмотреть, как это отразится на разбросе расстояний CA-CA. Исключаемых файлов

оказалось 815 (остался 2021 файл). Мы вычислили разброс расстояний CA-CA по этим 2021 файлу. Результат — разброс оказался равным 28%, т.е. причина такого большого разброса устранена не была.

Думаем, многие уже давно догадались, что реальной причиной является наличие как *цис*, так и *транс* конфигураций у пептидных групп, см. рис. 22.

2.5.1 Цис и Транс конфигурации пептидных групп

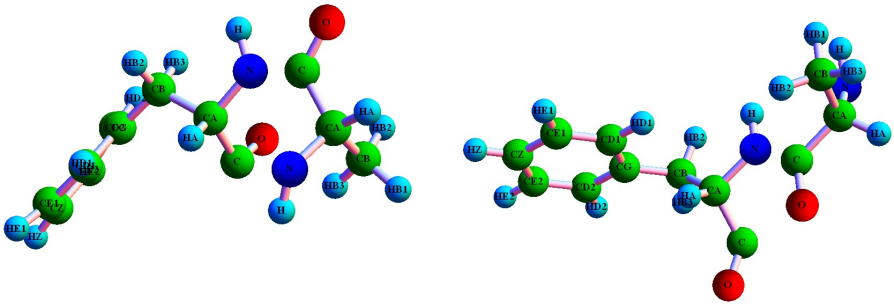

Рис. 22: Цис (слева) и транс (справа) конфигурации пептидных групп. В цис-конфигурации атомы CA находятся “с одной стороны”, а в транс-конфигурации — с разных.

Чтобы определить такие конфигурации формально, мы рассмотрели векторные произведения [N-CA, N-H] и [C-CA, C-O] для пептидной группы (в случае пролина вместо H берем CD) и назвали такую шестерку находящейся в *транс конфигурации*, если угол между этими векторами острый (их скалярное произведение положительно). В противном случае, мы отнесли эту шестерку к *цис конфигурациям*.

Мы посчитали количество полипептидов, в которых присутствуют последовательные пары аминокислот, находящиеся в цис-конфигурации. Оказалось, что таких полипептидов относительно мало, а именно, 263. Сначала мы попробовали, не улучшая нашу базу отсеиванием пептидных групп с большими объемами выпуклых оболочек, просто выбросить эти 263 файла и проверить, какой разброс будет у расстояний CA-CA. Оказалось, что разброс стал лучше, но не намного, а именно, он стал равен 17.67%. Тогда мы решили также выбросить полипептиды с наиболее значительными отклонениями от закона плоскости, а именно, все полипептиды с объемами выпуклых оболочек пептидных групп, превышающими 3. Оказалось, что мы должны избавиться от следующих 19 полипептидов:

`1AW3, 1BLV, 1EGO, 1ESK, 1FDF, 1I1S, 1I6C, 1L6H, 1MEA, 1MED,`
`1MNY, 1OZO, 1PBA, 1W9R, 1YUS, 2ADX, 2H3J, 2ITH, 7HSC.`

Результаты нас порадовали: теперь разброс длин CA-CA оказался равен 7.4%. В результате наша база стала содержать 2563 файла.

Наконец, мы добрались до проверки разброса углов между ковалентными связями.

2.6 Оценка разброса углов в полипептидах

Мы очень надеялись, что процедура проверки углов между ковалентными связями будет “чисто формальной”: ведь у нас уже отобраны полипептиды, в аминокислотах которых разбросы длин этих связей достаточно маленькие. Конечно, с формальной точки зрения, отсюда ничего не следует, но все-таки мы существенно “почистили” базу, поэтому есть шанс, что все большие отклонения уже были выкинуты. Тем не менее, нас ждал сюрприз: максимальное отклонение от среднего значения углов оказалось равным 71.1204%. На рис. 23 показаны три примера, соответствующих трем самым большим значениям разброса углов (мы подписали название аминокислоты и содержащего ее полипептида).

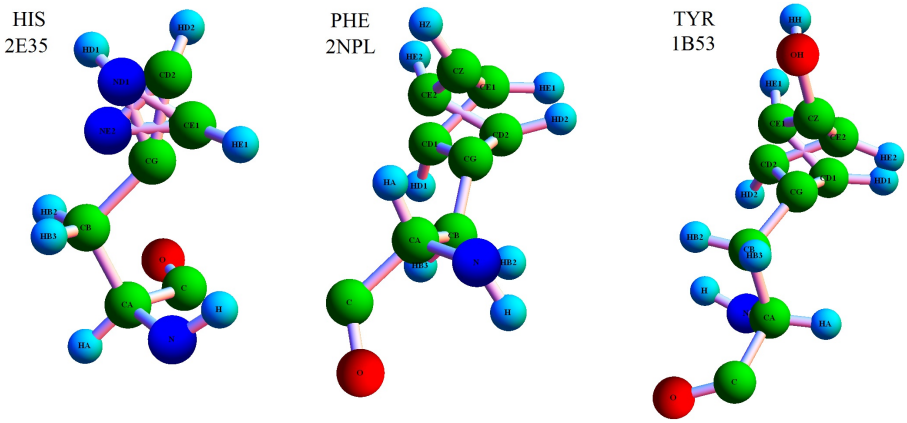

Рис. 23: Аминокислоты, в которых разброс углов принимает три самых больших значения.

Отметим, что в остальных аминокислотах, соответствующих меньшим разбросам, отклонения обусловлены существенно неравномерным распределение ковалентных связей для некоторых атомов валентности 4: соседи таких четырехвалентных атомов могут оказаться в одном полупространстве относительно некоторой плоскости, проведенной через этот атом, рис. 24.

Мы выяснили, что подобные сильные отклонения начинаются при разбросах величины около 26%. Всего же полипептидов с максимальным разбросом, превосходящим 26%, имеется 59 штук. Опять же, исходя из соображений “типичности”, мы исключили эти файлы из нашей базы, тем самым, оставив в базе 2532 файлов. Для полученной базы угловые отклонения в аминокислотах не превосходят 26%, метрические отклонения в аминокислотах не превосходят 10%, отклонения в расстояниях между последователь-

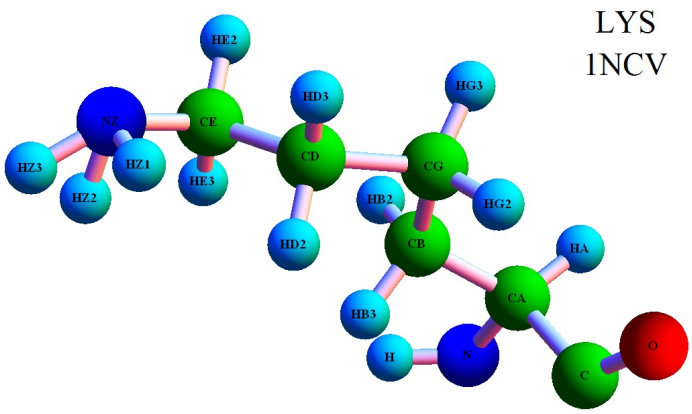

Рис. 24: Атомы, прикрепленные к NZ, лежат в одном полупространстве относительно некоторой плоскости, проходящей через NZ.

ными альфа-углеродами не больше 7.5%. Кроме того, все пептидные группы образуют транс конфигурации, и почти все пептидные группы достаточно точно удовлетворяют закону плоскости. Дальнейшее исследование будем проводить именно на этой базе.

3 Подвижность аминокислот

Рассматривая различные картинки с изображениями одноименных аминокислот в разных местах одного полипептида или в разных полипептидах, мы обнаружили, что эти аминокислоты сильно отличаются друг от друга. Для сравнительного анализа мы решили изобразить одноименные аминокислоты, встреченные в разных местах полипептидов, на одной картинке. Для этого нам нужно было придумать, как совмещать различные, но одноименные кислоты. Опираясь на анализ метрических соотношений, мы ввели понятие *репера аминокислоты*, а именно, в качестве *начала координат* мы выбрали альфа-углерод CA, а в качестве концов базисных векторов — азот N, углерод C и бета-углерод CB — последний во всех случаях, кроме "вырожденной" аминокислоты глицина; в глицине вместо CB мы выбрали водород HA3.

Замечание 3.1. Отметим, что в последовательности атомов аминокислоты, принятой в формате PDB, атомы N, CA, C, CB имеют номера соответственно 1, 2, 3, 5; в глицине же на 5-ом месте стоит совсем "не тот" водород (выбрав автоматически его, мы наблюдали поразительные картинки); нужный же нам водород в глицине расположен на 7-ом месте.

Имея реперы, естественно выбрать в качестве совмещающего преобразо-

вания одно из следующих двух: (1) аффинное преобразование, переводящее репер в репер; (2) ортогональное преобразование, располагающее соответствующие атомы реперов наиболее близким образом. Вот пример, рис. 25.

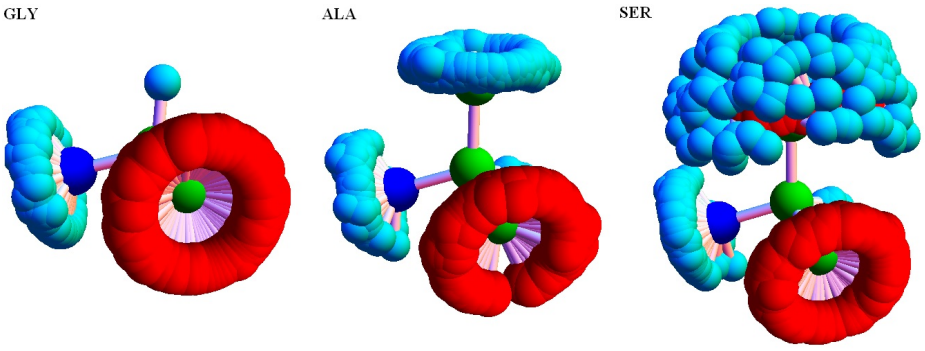

Рис. 25: Глицин, аланин, серин: совмещение реперов одноименных аминокислот.

Теперь хорошо видно, что прикрепленный к азоту N водород, а также прикрепленный к углероду C кислород могут вращаться вокруг оси соответствующего вектора репера. Также в случае аланина вращается и водород, прикрепленный к CB. Для серина ситуация более сложная: CB разветвляется на два водорода и кислород OG, к которому также крепится водород. Тройка, на которую разветвляется CB, может вращаться вокруг оси CA-CB; кроме того, прикрепленный к кислороду водород также может вращаться относительно оси CB-OG.

На рис. 26 и 27 приведены еще более сложные картинки.

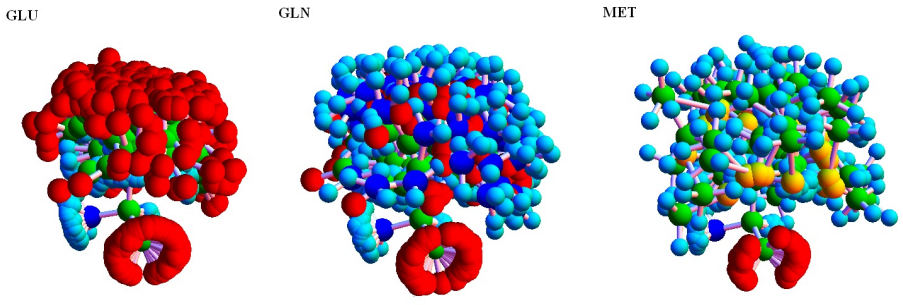

Рис. 26: Глутаминовая кислота, глутамин, метионин: совмещение реперов одноименных аминокислот.

Возможно, адекватное представление аминокислот, входящих в поли-

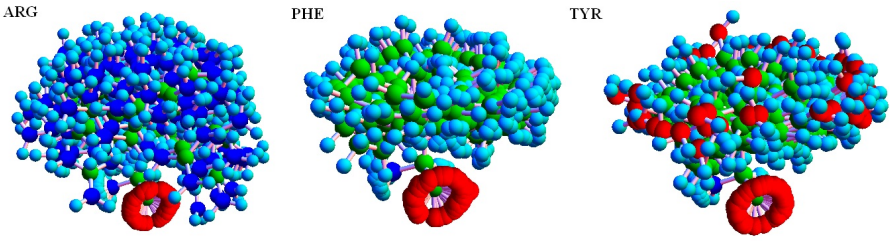

Рис. 27: Аргинин, фенилаланин, тирозин: совмещение реперов одноименных аминокислот.

пептид, состоит в рассмотрении соответствующих "облаков". Впрочем, водородные связи должны уменьшать *конфигурационное пространство* каждой аминокислоты. Было бы интересно разобраться, как выглядят эти конфигурационные пространства.

Замечание 3.2. При проведения описанного выше компьютерного эксперимента мы первоначально использовали преобразования, располагающие оптимальным образом атомы реперов, но сохраняющие при этом ориентацию этих реперов. В результате мы выяснили, что не *все реперы являются положительно ориентированными*. Мы решили в этом разобраться.

3.1 Ориентация аминокислот

Мы назвали аминокислоту *положительно ориентированной*, если ориентация ее репера совпадает со стандартной, иными словами, если определитель матрицы, строки которой — векторы репера, является положительным числом. В противном случае, назовем аминокислоту *отрицательно ориентированной*.

В последней нашей базе мы нашли все полипептиды, каждый из которых содержит хотя бы одну отрицательно ориентированную аминокислоту. Таких полипептидов оказалось 140. Взяв первый попавшийся из них, мы обнаружили там глицин, изображенный на рис. 28 слева.

Впрочем, если поменять местами названия водородов HA2 и HA2, то левый глицин станет положительно ориентирован, см. рис. 29.

Отметим, что глицинов, у которых репер с водородом HA3 положительно ориентирован, намного больше, чем глицинов, у которых положительно ориентирован репер с водородом HA2.

Таким образом, недоразумение с ориентацией в глицинах, в принципе, легко исправляется. Может, это — единственная причина возникновение неориентированных аминокислот? Ответ оказался отрицательным. Однако, количество полипептидов, содержащих отрицательно ориентированные аминокислоты, отличные от глицина, достаточно мало (равно четырем). Вот список имен этих полипептидов:

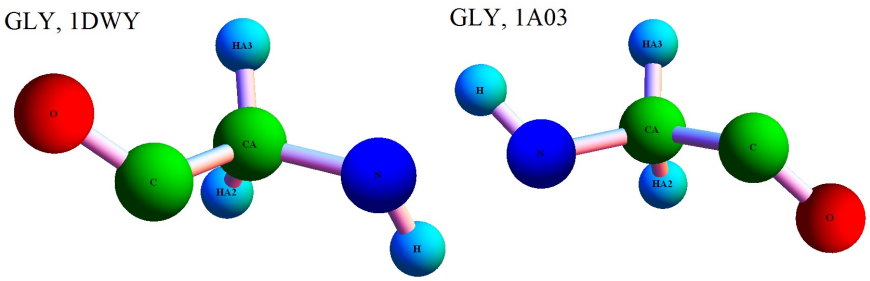

Рис. 28: Отрицательно ориентированный глицин (слева) и, для сравнения, положительно ориентированный глицин (справа).

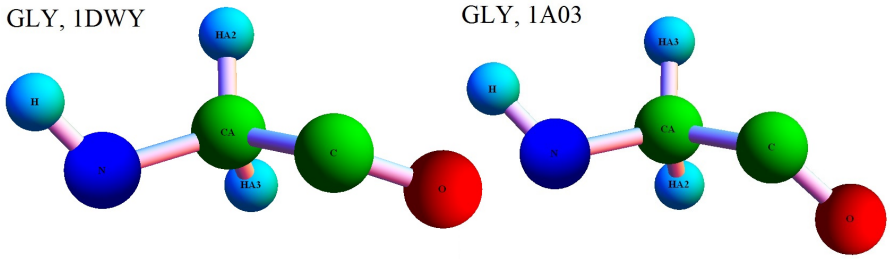

Рис. 29: Отрицательно ориентированный глицин (слева) и, для сравнения, положительно ориентированный глицин (справа).

`1NMJ, 1ZDB, 2B8F, 2NS4`

Отметим, что в каждом из таких полипептидов имеется ровно одна отрицательно ориентированная аминокислота: в трех их них она располагается на последнем месте, а в одном — на втором. Ниже мы приводим изображения этих аминокислот, вместе с их положительно ориентированными образцами из полипептида 1A03.

4 Добавление (совместно с Е.А.Вилкул)

Обсуждение со специалистами идей, изложенных выше, подняло ряд вопросов, которые мы хотим вкратце обсудить в настоящем добавлении. Напомним, что, желая работать с наиболее надежными данными, мы остановились на первой модели и выбрали из всех pdb-файлов те, в которых считающиеся устойчивыми метрические характеристики наиболее близки к их средним значениям. Попутно мы обнаружили разного рода “патологии”. Как заметили специалисты, одной из причин наблюдаемых нами патологий может являться тот факт, что первая модель бывает как раз не репрезента-

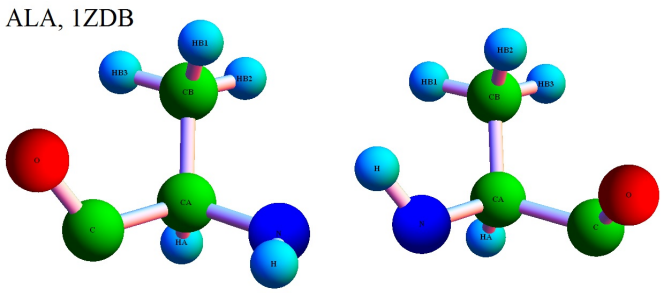

Рис. 30: Отрицательно ориентированный ALA (слева).

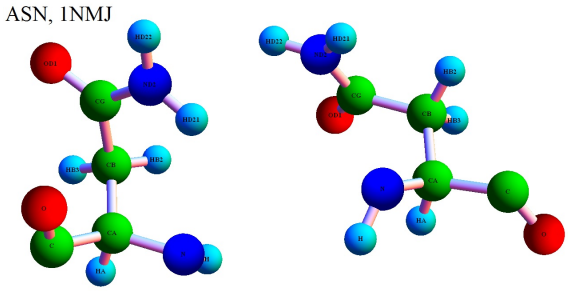

Рис. 31: Отрицательно ориентированный ASN (слева).

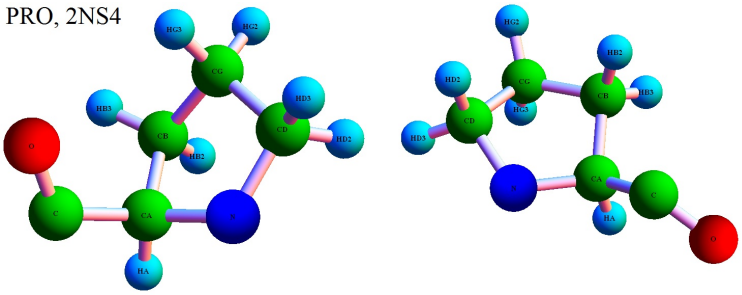

Рис. 32: Отрицательно ориентированный PRO (слева).

тивной. Мы решили посмотреть другие модели и выяснить, можно ли среди них найти более надежные. Детальный отчет мы опубликуем в другой раз, а здесь покажем лишь некоторые примеры.

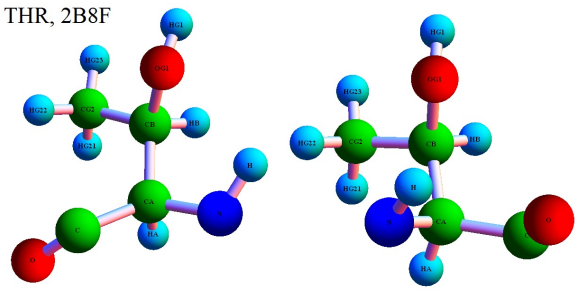

Рис. 33: Отрицательно ориентированный THR (слева).

4.1 Распределение количеств моделей

Прежде всего, разберемся, сколько моделей может встречаться в тех или иных pdb-файлах, каковы количества k-модельных файлов, и как связаны эти количества с годами, в которые проводились соответствующие исследования. “Существует мнение”, что, как правило, в pdb-файлах, описывающих структуры полипептидов, полученные с помощью ЯМР, моделей почти всегда достаточно много, в основном, порядка 15-20. Кроме того, файлы, содержащие мало моделей, скажем одну, датируются как правило прошлым веком.

Мы решили проверить эти утверждения. Для этого мы взяли 2892 файла, с которых мы стартовали в разделе 2, и вычислили, сколько в каждом из них содержится моделей, а также к каким годам относятся те или иные количества моделей.

Приведем сначала возможные количества моделей в тех или иных файлах. Вот их список:

```
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
52, 54, 55, 56, 58, 59, 60, 63, 64, 68, 70, 72, 76, 80, 81, 84,
90, 92, 96, 100, 108, 116, 120, 128, 140, 144, 160, 240, 290,
304, 308, 320, 640
```

Заметим, что максимальное число моделей равно 640, и оно встречается ровно в одном файле, а именно, в 2KOX.pdb. Этот файл был получен в 2009 году. Следующие “рекордсмены” содержат по 320 моделей; это — два файла 2K0F.pdb и 2M5N.pdb, первый из которых получен в 2008 году, а второй — в 2013.

Затем мы вычислили, сколько имеется файлов с тем или иным количеством моделей:

```
{1, 315}, {2, 50}, {3, 7}, {4, 9}, {5, 22}, {6, 5}, {7, 3},
{8, 3}, {9, 12}, {10, 240}, {11, 20}, {12, 24}, {13, 9},
```

```
{14, 21}, {15, 82}, {16, 22}, {17, 9}, {18, 24}, {19, 25},
{20, 1291}, {21, 40}, {22, 17}, {23, 14}, {24, 23}, {25, 58},
{26, 10}, {27, 5}, {28, 7}, {29, 11}, {30, 118}, {31, 13},
{32, 22}, {33, 8}, {34,  10}, {35, 16}, {36, 13}, {37, 1},
{38, 8}, {39, 4}, {40, 164}, {41, 4}, {42, 11}, {43, 2},
{44, 7}, {45, 2}, {46, 5}, {47, 3}, {48, 3}, {49, 2}, {50, 18},
{52, 2}, {54, 1}, {55, 2}, {56, 3}, {58, 2}, {59, 1}, {60, 20},
{63, 1}, {64, 1}, {68, 2}, {70, 3}, {72, 1}, {76, 1}, {80, 15},
{81, 1}, {84, 1}, {90, 1}, {92, 1}, {96, 1}, {100, 3}, {108, 1},
{116, 1}, {120, 4}, {128, 1}, {140, 1}, {144, 1}, {160, 1},
{240, 1}, {290, 1}, {304, 1}, {308, 1}, {320, 2}, {640, 1}
```

(здесь в каждой паре на первом месте стоит количество моделей, а на втором — количество файлов с таким количеством моделей). Видно, что при каждом $n > 80$ число файлов, содержащих ровно n моделей, не превосходит 4 и, почти всегда, равно 1. Мы выкинули все такие файлы, а для оставшихся изобразили график, по абсциссе которого отложено число моделей (от 1 до 80), а по ординате — количество файлов с таким числом моделей, см. рис. 34.

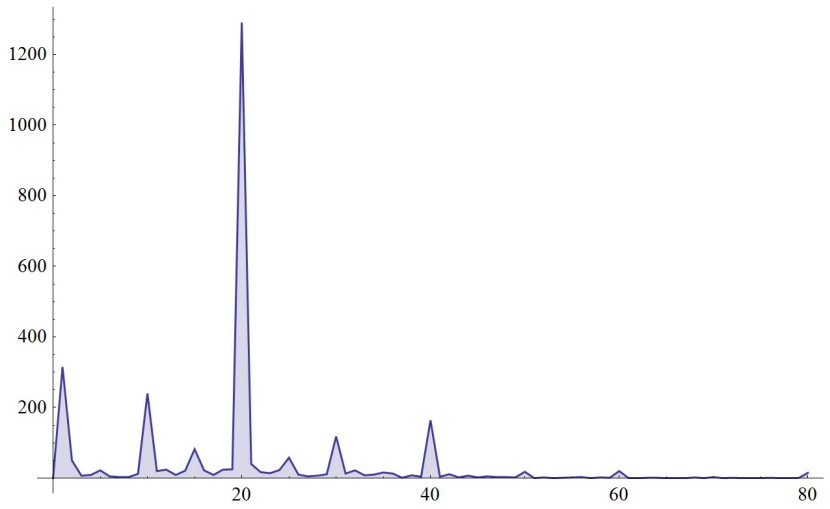

Рис. 34: Распределение количеств файлов, содержащих те или иные количества моделей, находящиеся в пределах от 1 до 80.

На графике хорошо видно, что максимальное число файлов приходится на количество моделей, равное 20: такое количество моделей содержится в 1291 файле. Кроме того, видно также, что 1-модельных файлов тоже достаточно много, а именно, 315 (это — следующий максимум).

Построим распределение 20-модельных файлов по годам, см. рис. 35.

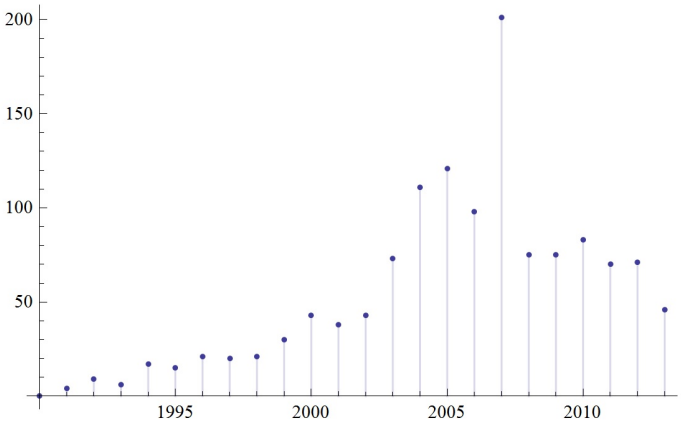

Рис. 35: Распределение по годам количеств 20-модельных файлов.

Видно, что максимальное количество приходится на 2007 год (201 файл), хотя и в 1993 году произвели 4 таких файла.

Наконец, посмотрим распределение по годам количеств файлов, содержащих по 1 модели, рис. 36 (всего имеется 315 таких файлов).

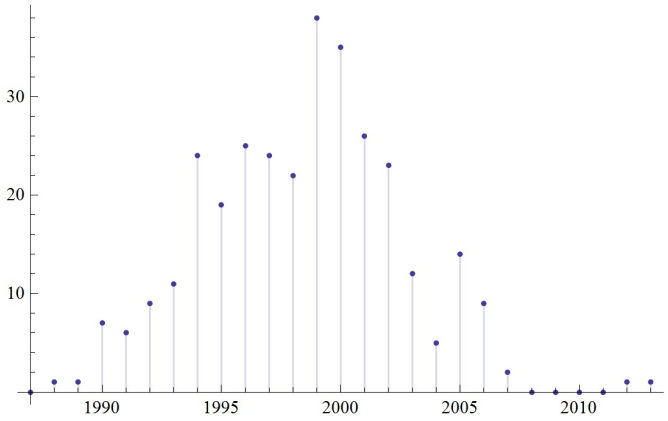

Рис. 36: Распределение по годам количеств 1-модельных файлов.

Как видно, максимальное число 1-модельных файлов появилось в 1999-2000 годах (в сумме там имеется 73 файла), а, начиная с 2005 года, было создано 27 таких файлов.

4.2 “Репрезентативность” первой модели в контексте патологий

В предыдущих разделах мы обнаружили ряд “патологических” файлов, в которых координаты атом вычислены заведомо неправильно. Кроме этого, мы обнаружили ряд других отклонений, например, от закона плоскости. Как нам было сказано, одной из возможных причин может служить то, что выбранная нами первая модель часто бывает не репрезентативна. Мы решили проверить это утверждение.

Напомним, что в разделе 2.1 мы изучали относительные отклонения от среднего длин ковалентных связей, и обнаружили, что в некоторых случаях эти отклонения могут быть более 700%. Мы установили, что этот “результат” достигается на файле 2PDE.pdb. Может быть, всему виной первая модель, а на остальных моделях таких отклонений нет? Увы, этот файл содержит только одну модель!

Тогда мы решили изучить все 30 файлов, в каждом из которых имеются более 15%-ые относительные отклонения длин ковалентных связей от среднего значения (список этих файлов был приведен в разделе 2.1). Мы выяснили, что в 12 из них имеется только одна модель, так что никаких улучшений в этих 12 файлах сделать не удастся.

В оставшихся 18 файлах количества моделей распределены так:

```
{9,1},{10,3},{11,1},{16,2},{20,7},{21,1},{34,1},{35,1},{81,1}
```

Будем “идти с конца”, т.е. рассматривать оставшиеся файлы с максимальным числом моделей. Среди них будем брать те, в которых патология очевидна. Первым из таких файлов оказывается 1MZI, содержащий 81 модель. В его первой модели отклонения не привели к заметному нарушению канонического вида его аминокислот.

Вторым (с 35 моделями) оказался файл 1OT4, в котором триптофан (TRP), находящийся на 83 месте (82 внутренняя аминокислота), имеет очевидное нарушение. Просмотрев все оставшиеся модели, мы выяснили, что это нарушение сохраняется во всех 35 моделях, см. рис. 37.

Заметим, что триптофан на приведенных картинках выглядит очень похоже. Может, это просто продублированная первая модель? Мы проверили координаты одноименных атомов триптофана в разных моделях: оказалось, что они — разные.

Третий пример — файл 2I2J с 34 моделями. В первой его модели все аминокислоты даже на глаз выглядят неправильно, см. рис. 8. Мы просмотрели все остальные модели: в них, как оказывается, уже все в порядке (все аминокислоты выглядят стандартным образом), так что этот файл подтверждает мнение специалистов.

Четвертый пример аналогичен второму, и его дает файл 1KWD (содержащий 16 моделей): 9-ая внутренняя аминокислота имеет неправильный вид также во всех моделях, см. рис. 38.

В дальнейшем, мы проведем детальный анализ всех “устойчивых” патологий (т.е. присутствующих во всех моделях).

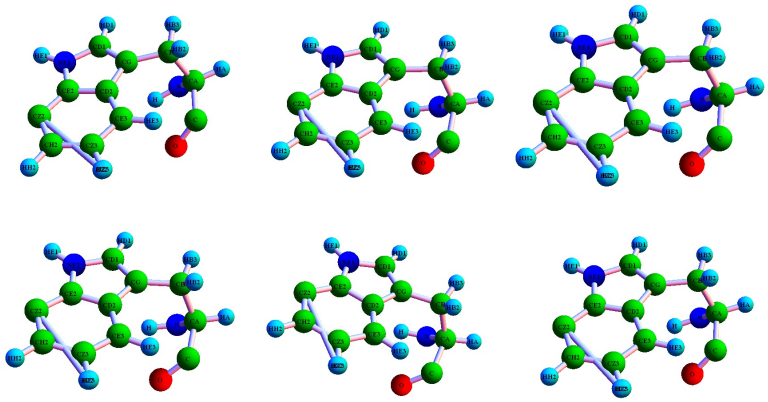

Рис. 37: Вид “патологического” триптофана в первых 6 моделях (в остальных 29 — такой же).

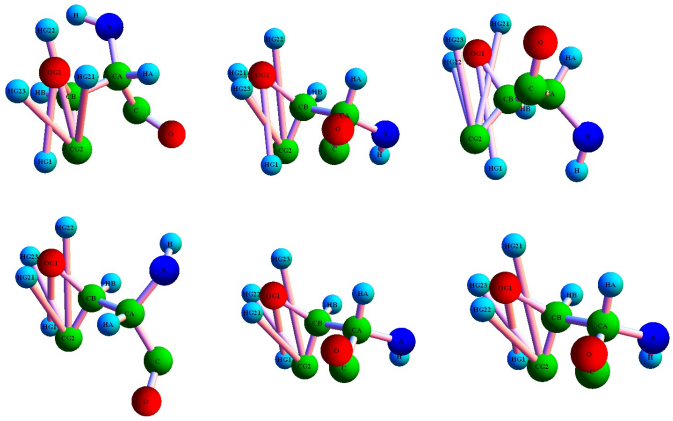

Рис. 38: Вид “патологического” аланина в первых 6 моделях.

4.3 “Репрезентативность” первой модели в контексте закона плоскости

В разделе 2.4 мы, на примере первой модели, показали, что во многих случаях нарушается закон плоскости. Посмотрим, что же происходит с другими моделями.

Напомним, что для измерения того, насколько нарушается закон плоскости, мы вычисляли объем выпуклой оболочки, натянутой на пептидную группу. Значения объема, не большие 0.5, соответствовали хорошему вы-

полнению закона; отклонения больше 1 уже хорошо заметны на глаз; максимальные отклонения были порядка 4.5, см. выше.

Мы решили проанализировать "рекордсменов–нарушителей"закона плоскости. Напомним, что первое место занял файл 1PBA. В этом файле содержится 20 моделей. Для наглядности мы решили по каждой модели построить график объемов для последовательных пар аминокислот. Вот, что получилось для первой модели, см. рис. 39.

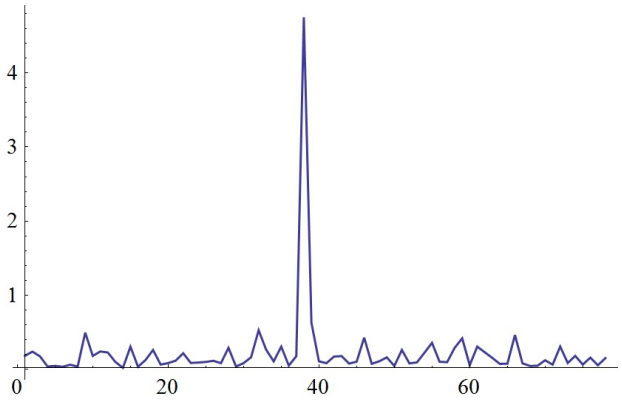

Рис. 39: Объемы, измеряющие отклонение от закона плоскости, для последовательных пар аминокислот первой модели полипептида 1PBA.

Хорошо видно, что в первой модели этого полипептида имеется ровно одно серьезное отклонение от закона плоскости. Мы будем называть его *большой пик*. Он возникает на паре аминокислот с номерами (38,39).

А вот, что происходит во второй модели, см. рис. 40.

На графике отчетливо видно, что, помимо большого пика, появился второй, *маленький пик*, величина которого больше 2, а это значит, что соответствующее отклонение существенно. Этот пик соответствует паре (46,47).

Приведем еще ряд графиков, см. рис. 41.

Заметим, что на этих двух графиках отклонение меньше 1, так что закон плоскости выполняется более-менее. Обратите внимание, что на левом графике присутствуют оба пика, а на правом “большой” пик отсутствует, а имеется только “маленький”.

Отметим также, что на 14-ти моделях отклонение больше 2, а на 5-ти моделях отклонение меньше 1. Кроме того, самый большой пик всегда возникает или на паре (38,39), или на паре (46,47).

Приведем еще один пример. Он интересен тем, что у него существенно больше серьезных отклонений (с объемом больше 4). Для этого построим соответствующие графики для файла 1I1S (содержит 20 моделей), см. рис. 42.

Попробуем найти закономерности в нарушении закона плоскости. Как

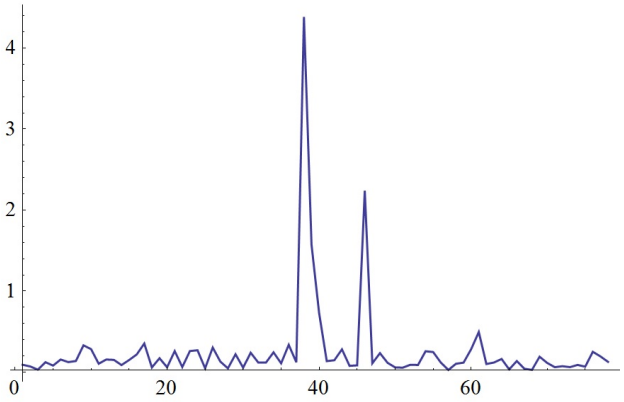

Рис. 40: Объемы, измеряющие отклонение от закона плоскости, для последовательных пар аминокислот второй модели полипептида 1PBA.

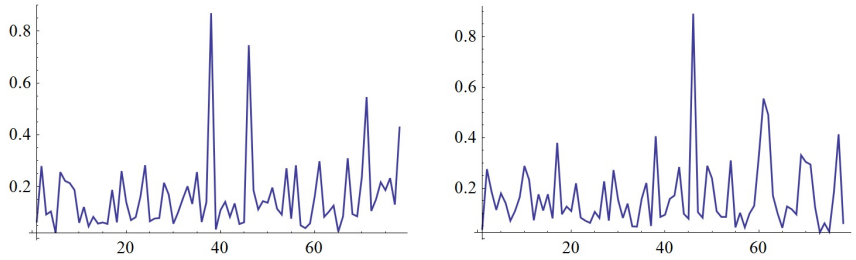

Рис. 41: Объемы, измеряющие отклонение от закона плоскости, для последовательных пар аминокислот 6-ой и 12-ой моделей полипептида 1PBA.

известно, типичными элементами вторичной структуры белка являются спирали и листы. Информацию об их расположении в полипептиде можно извлечь из самого PDB-файла (строки, начинающиеся со слов HELIX и SHEET соответственно).

Для удобства на каждом из графиков отметим на оси, соответствующей номерам аминокислот, отрезки спиралей красным цветом, а листов — синим. Заметим, что единственный пик первой модели файла 1PBA, соответствующий паре аминокислот (38,39), попал вне выделенных областей, см. рис. 43.

Похожая ситуация наблюдается и для второй модели: здесь маленький пик, возникший на паре (46,47), попал на границу одного из отрезков. Также можно заметить некое "затухание", происходящее при движении вдоль горизонтальной оси от левого конца, где располагается большой пик, к середине второго красного отрезка, см. рис. 44.

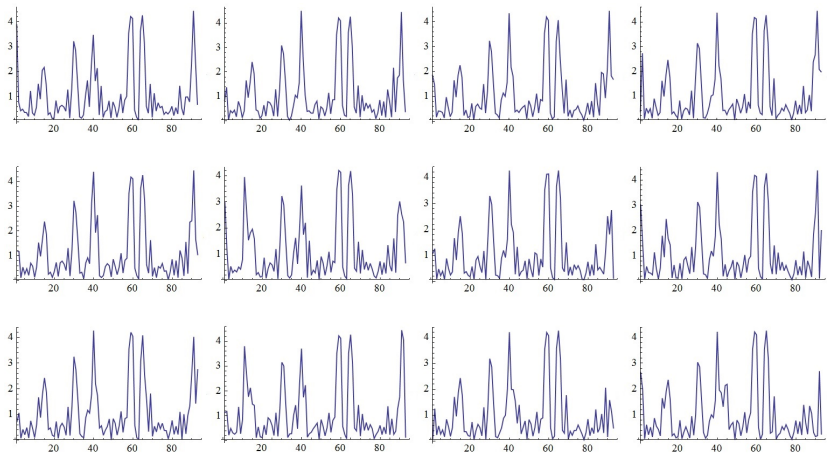

Рис. 42: Объемы, измеряющие отклонение от закона плоскости, для последовательных пар аминокислот первых 12-ти моделей полипептида 1I1S.

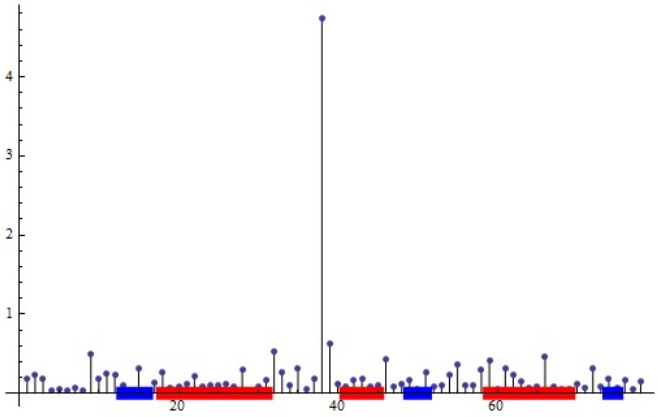

Рис. 43: График объемов для последовательных пар аминокислот первой модели полипептида 1PBA с выделенными отрезками спиралей и листов.

Приведем такие же графики и для полипептида 1I1S (здесь наблюдается намного больше пиков). При внимательном рассмотрении можно увидеть ту же закономерность: самые большие отклонения от закона плоскости приходятся либо на границу отрезков спиралей и листов, либо возникают вне их, а при движении от концов отрезков к их серединам происходит "затухание" эффекта, см. рис. 45.

Данный феномен может быть обоснован тем, что спирали и листы —

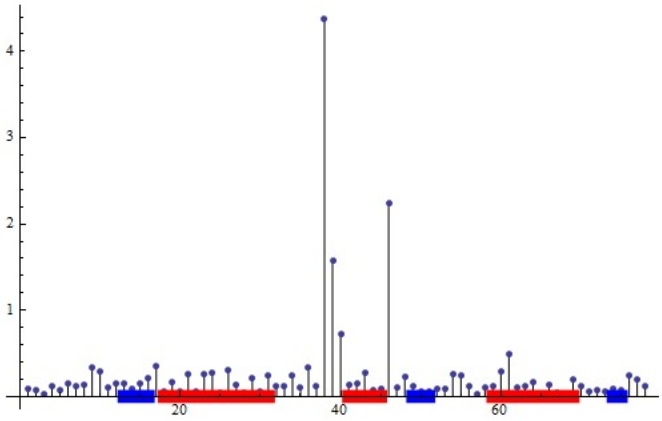

Рис. 44: График для второй модели полипептида 1PBA.

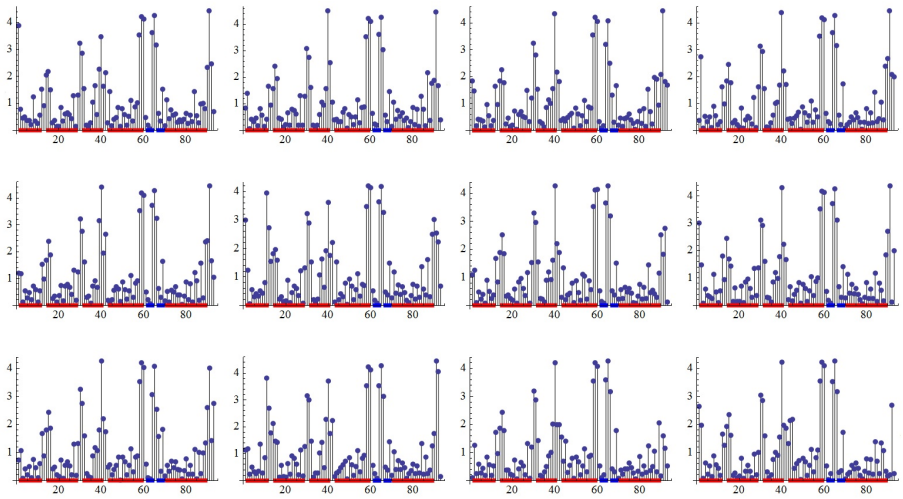

Рис. 45: Объемы, измеряющие отклонение от закона плоскости, для последовательных пар аминокислот первых 12-ти моделей полипептида 1I1S с выделенными отрезками спиралей и листов.

достаточно устойчивые структуры, а промежутки между ними наоборот подвижны, что и приводит к стабилизации возникших отклонений по мере движения вдоль типичных элементов вторичной структуры белка.

5 Геометрия плоских и пространственных ломаных. Автоматическое выделение спиралей

В этом разделе мы приведем еще ряд геометрических идей, которые могут оказаться полезными при исследовании конформации полипептидов. Рассмотрим произвольную *геометрическую реализацию* полипептида, т.е. некоторое конкретное расположение его в пространстве. Начнем с изучения расположения альфа-углеродов. Упорядочив эти углероды в соответствии с порядком аминокислот, мы можем представлять их вершинами некоторой ломаной L. Что нужно знать про ломаную, чтобы по имеющимся данным ломаная восстанавливалась бы однозначно с точность до сохраняющего ориентацию движения? Ответ более менее очевиден: нужно знать длины ребер, углы поворота между предыдущим и последующим ребром, а также углы поворота между каждой парой плоскостей, первая из которых натянута на $(i-1)$-ое и i-ое ребра, а вторая — на i-ое и $(i+1)$-ое.

Давайте более строго определим, что понимается под описанными только что углами. Пусть $L = A_0A_1 \cdots A_n$, где $A_i \in \mathbb{R}^3$, — *вершины* ломаной. *Ребро* e_i *ломаной* L — это пара $A_{i-1}A_i$. Нам будет удобно представлять ребра e_i как векторы $A_i - A_{i-1}$. Мы будем всегда предполагать, что последовательные векторы e_i, e_{i+1} неколлинеарны. Пусть α_i — величина угла между векторами e_i, e_{i+1}, т.е.

$$\alpha_i = \arccos \frac{\langle e_i, e_{i+1}\rangle}{\|e_i\| \cdot \|e_{i+1}\|},$$

где $\langle v, w\rangle$ — стандартное скалярное произведение векторов v и w, а $\|v\|$ — соответствующая норма вектора, т.е. $\|v\| = \sqrt{\langle v, v\rangle}$. Из сделанного предположения вытекает, что $0 < \alpha_i < \pi$.

Рассмотрим теперь произвольное *внутреннее* (т.е. неконцевое) ребро e_i ломаной L, тогда имеются соседние с ним ребра e_{i-1} и e_{i+1}. Мы хотим определить величину β_i угла между плоскостями, первая из которых натянута на ребра e_{i-1}, e_i, а вторая — на e_i, e_{i+1}. Мы поступим несколько хитрее, а именно, будем определять углы между этими плоскостями, но естественным образом *ориентированными*.

Более подробно, рассмотрим векторные произведения $\xi_i = [e_{i-1}, e_i]$ и $\xi_{i+1} = [e_i, e_{i+1}]$. Первый из этих векторов перпендикулярен первой плоскости, а второй — второй. Рассмотрим плоскость Π, перпендикулярную e_i. Тогда оба ξ_i и ξ_{i+1} ей параллельны. Введем в плоскости Π декартовы координаты. Для этого в качестве начала координат выберем точку пересечения плоскости Π и прямой, проходящей через ребро e_i; ось x пустим в направлении e_x вектора ξ_i; ось y выпустим в таком направлении e_y, чтобы тройка ξ_i, e_y и e_i была положительно ориентирована. Иными словами, $e_y = \big[e_i/\|e_i\|, \xi_i/\|\xi_i\|\big]$. Теперь рассмотрим направление $e = \xi_{i+1}/\|\xi_{i+1}\|$ вектора ξ_{i+1}. Тогда в базисе e_x, e_y вектор e имеет координаты $(\cos\beta_i, \sin\beta_i)$ для некоторого однозначно определенного угла $\beta_i \in [-\pi, \pi)$ (именно его мы будем использовать в качестве угла между рассматриваемыми плоскостя-

ми).

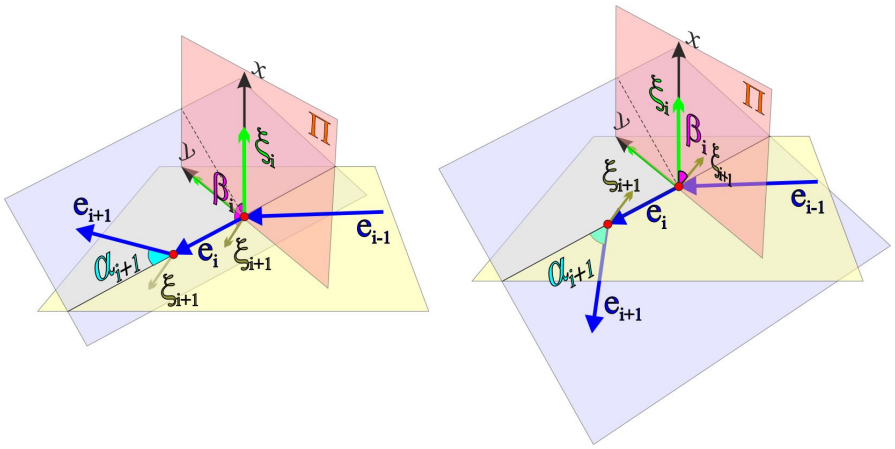

Рис. 46: Угловые характеристики ломаной.

Найдем угол β_i явно. Заметим, что $\cos\beta_i = \langle e, e_x\rangle$, $\sin\beta_i = \langle e, e_y\rangle$, поэтому

$$\beta_i = \operatorname{sign}(\sin\beta_i)\arccos(\cos\beta_i) = \operatorname{sign}\big(\langle e, e_y\rangle\big)\arccos\big(\langle e, e_x\rangle\big) = \\ = -\operatorname{sign}\big[\det(\xi_i, e_i, \xi_{i+1})\big]\ \arccos\frac{\langle \xi_i, \xi_{i+1}\rangle}{\|\xi_i\|\,\|\xi_{i+1}\|}.$$

Покажем, что длины s_i ребер ломаной L ее углы α_i, β_j определяют ломаную L однозначно с точностью до движения.

(1) Выберем произвольное расположение начальной вершины A_0 ломаной L.

(2) Разместим вторую вершину A_1 ломаной L произвольным образом на расстоянии s_1.

(3) Выпустим из вершины A_1 луч r_1 под углом α_1 к лучу $r_0 = A_0A_1$ (это можно сделать многими способами) и отложим на r_1 отрезок A_1A_2 длины s_2.

Если у каждого из предыдущих построений была некоторая свобода выбора, то все оставшиеся шаги будут определены однозначно. Мы покажем это для шага, на котором строится вершина A_{i+1}. Итак, пусть все вершины $A_0, \dots, A_i$ уже построены. В частности, у ломаной L уже реализованы ребра e_{i-1} и e_i.

(4) Вычисляем вектор $\xi_i = [e_{i-1}, e_i]$ и полагаем $e_x = \xi_i/\|\xi_i\|$.

(5) Вычисляем вектор $e_y = [e_i, e_x]/\|e_i\|$.

(6) Тогда, по определению, имеем $\xi_{i+1}/\|\xi_{i+1}\| = \cos\beta_i\, e_x + \sin\beta_i\, e_y$.

(7) Чтобы задать положение точки A_{i+1}, мы должны в плоскости, проходящей через ребро e_i и перпендикулярной ξ_{i+1}, отложить от A_i вектор

e_{i+1} длины s_{i+1} в направлении, составляющем с вектором e_i угол α_{i+1}. Это можно сделать двумя способами. Однако, по определению вектора ξ_{i+1}, он должен быть равен $[e_i, e_{i+1}]$, так что подходит лишь один из этих двух способов. А именно,

$$e_{i+1} = s_{i+1}\left(\cos\alpha_{i+1}\,\frac{e_i}{\|e_i\|} + \sin\alpha_{i+1}\left[\cos\beta_i\,e_x + \sin\beta_i\,e_y,\ \frac{e_i}{\|e_i\|}\right]\right).$$

Пусть теперь все ребра ломаной L имеют одинаковую длину. Тогда постоянство отличных от нуля и π углов α_i и β_i равносильно тому, что вершины ломаной лежат на некоторой спирали. Это соображение позволяет отлавливать спирали в полипептидах. Напомним, что в оставшейся после очистки базе данных во всех полипептидах расстояния между последовательными альфа-углеродами отклоняются от среднего значения не более, чем на 7.5%, поэтому эти расстояния можно считать почти постоянными. Таким образом, если для ломаной L, проведенной через альфа-углероды, на каком-то ее фрагменте углы α_i и β_i окажутся почти постоянными (и отличными от 0 и π), то этот фрагмент будет являться спиралью. Продемонстрируем, как это работает на примере.

Рассмотрим полипептид 1A03. Для наглядности мы решили через альфа-углероды этого полипептида провести кривую. На рис. 47 (слева) хорошо видно, что этот полипептид имеет четыре спирали.

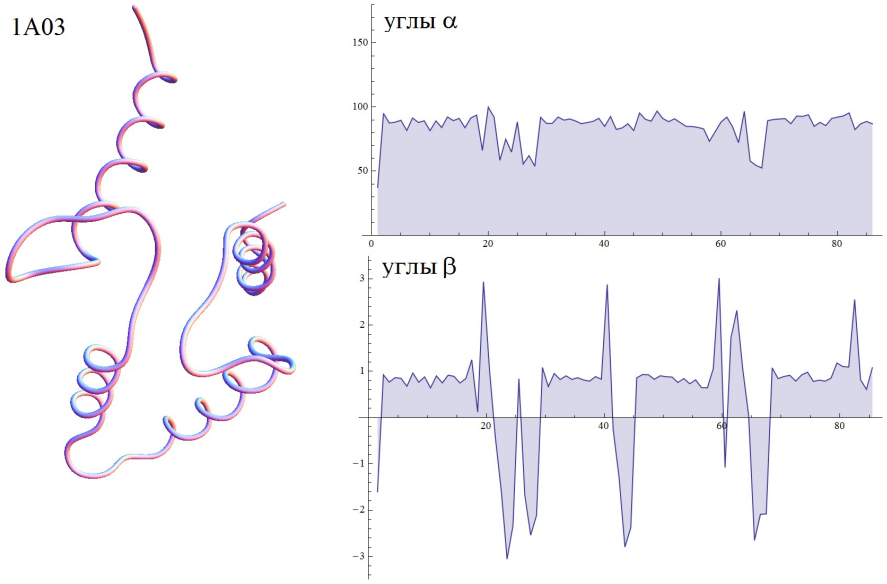

Рис. 47: Аппроксимация пептидного остова полипептида 1A03 кривой (слева) и графики последовательных углов α_i и β_i (справа).

Справа на этом рисунке приведены последовательности α- и β-углов ломаной, вершины которой — последовательные альфа-углероды полипептида 1A03. На этих графиках хорошо просматриваются участки локального постоянства. Таких участков также имеется четыре штуки, и они соответствуют спиралям полипептида. Несложно написать программу, которая будет автоматически определять участки постоянства графиков углов, тем самым, выделять спирали программным способом, не прибегая к изображению.

Для большей наглядности можно изобразить оба графика одновременно (растянув для наглядности один из них), рис. 48.

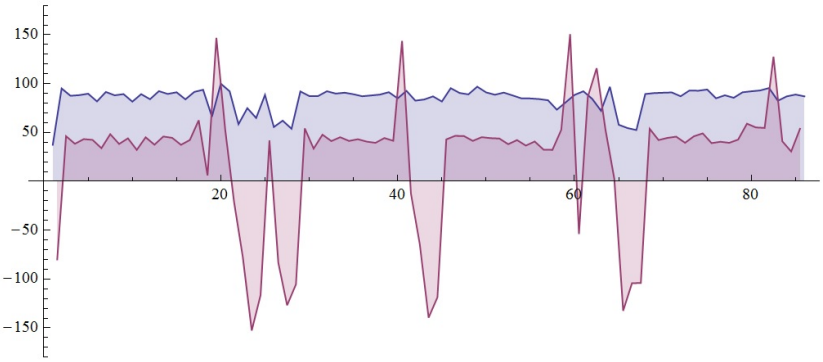

Рис. 48: Одновременное изображение обоих графиков α- и β-углов.

Замечание 5.1. Было бы интересно найти критерий того, что произвольная ломаная (не обязательно с равными звеньями) вписана в спираль. Это условие позволило бы проводить аналогичное автоматическое определение спиралей и для полипептидов с большими отклонениями расстояний между последовательными альфа-углеродами.

В чем преимущество описанного только что подхода? Для коротких полипептидов можно, конечно же, определять фрагменты структуры и на глаз, но если полипептид достаточно длинный, то непосредственное изучение его трехмерного изображения является достаточно тяжелой работой. Вместо этого мы предлагаем смотреть на две функции и исследовать их (или на три, если учитывать расстояния). Кроме того, различные свойства участков функций, скажем, монотонность или выпуклость, могут привести к целому ряду новых типов фрагментов полипептидов и, как следствие, к более глубокому пониманию геометрической структуры.

На самом деле, изложенные здесь соображения характеризации спиралей в терминах углов являются дискретными аналогами хорошо известных в дифференциальной геометрии результатов, описывающих форму пространственных кривых. В следующем разделе мы немного поговорим и об этом, чтобы предложить читателю еще ряд тем для поиска аналогий.

6 Кривизна и кручение пространственных кривых

Рассмотрим гладкую кривую $\gamma(t) = \big(x_1(t),\, x_2(t),\, x_3(t)\big)$ в пространстве $\mathbb{R}^3$. Здесь гладкость означает, что координатные функции $x_i(t)$ непрерывно-дифференцируемы столько раз, сколько нам потребуется. *Скоростью кривой γ в точке $\gamma(t)$* называется вектор $\dot\gamma(t) = \big(\dot x_1(t),\, \dot x_2(t),\, \dot x_3(t)\big)$, составленный из первых производных координатных функций. Если вместо первых производных рассмотреть вторые производные, то получим *вектор ускорения* $\ddot\gamma(t) = \big(\ddot x_1(t),\, \ddot x_2(t),\, \ddot x_3(t)\big)$. Кривая, у которой скорость всюду отлична от нуля, называется *регулярной*, а у которой скорость и ускорение в каждой точке линейно независимы — *бирегулярной*.

Если длина вектора скорости кривой $\gamma(t)$ всюду равна 1, то параметр t называется *натуральным*. Для натурально параметризованной кривой скорость и ускорение взаимно перпендикулярны в каждой точке кривой. Для такой кривой длина ускорения используется для характеризации искривленности: чем больше ускорение, тем сильнее кривая поворачивает. Длина вектора ускорения натурально параметризованной кривой называется *кривизной* и обозначается через k. Если кривизна отлична от нуля, т.е. ускорение ненулевое и, значит, кривая бирегулярна, то направление $\ddot\gamma/\|\ddot\gamma\|$ вектора ускорения $\ddot\gamma$ называется *главной нормалью кривой* γ и обозначается через ν.

Итак, пусть кривая $\gamma(t)$ натурально параметризована. Положим $\tau = \dot\gamma$, тогда τ и ν — единичные взаимно перпендикулярные векторы. Их векторное произведение $\beta = [\tau, \nu]$ называется *бинормалью*. Тройка (τ, ν, β) единичных взаимно перпендикулярных векторов называется *репером Френе*.

Формулы Френе показывают, с какими скоростями меняются векторы репера Френе. Известно, что скорость $\dot\beta$ изменения бинормали β коллинеарна вектору ν. Коэффициент пропорциональности между $\dot\beta$ и ν, взятый со знаком минус, называется *кручением* и обозначается через $\varkappa$.

Приведем некоторые примеры, показывающие, как свойства кривизны и кручения выделяют те или иные естественные классы кривых.

Пример 6.1. Множество всех прямых — это в точности множество всех кривых нулевой кривизны (отметим, что для прямых кручение не определено).

Пример 6.2. Кривизна окружности радиуса r равна $1/r$. Лежащая в плоскости кривая является окружностью, если и только если ее кривизна постоянна и отлична от нуля.

Пример 6.3. У *винтовой линии* или *спирали*, которая в некоторой системе координат имеет вид $(a\cos t,\ a\sin t,\ b\,t)$, где a и b — ненулевые постоянные, кривизна и кручение постоянны и не равны нулю. Обратно, каждая бирегулярная кривая, у которой кривизна и кручение постоянны и отличны от нуля, является винтовой линией.

Пример 6.4. Бирегулярная кривая лежит в некоторой плоскости, если и только если ее кручение равно нулю.

Хорошо известно, что функции $k(t)$ и $\varkappa(t)$ полностью определяют форму натурально параметризованной бирегулярной кривой $\gamma(t)$: если у двух кривых эти функции одинаковы, то кривые можно совместить. Более того, каждые две функции $f(t)$ и $g(t)$, где $f(t)$ везде положительна, являются кривизной и кручением некоторой натурально параметризованной бирегулярной кривой. Процедура построения пространственной кривой с заданными кривизной и кручением сводится к решению некоторой системы обыкновенных дифференциальных уравнений.

В действительности, характеризация ломаных в терминах углов — это "дискретный аналог" характеризации кривых их кривизнами и кручениями. При этом условие на равенство длин ребер ломаной аналогично натуральности параметра.

Если на кривой $\gamma(t)$ параметр t не натуральный, то кривизну и кручение можно вычислить по следующим формулам:

$$k(t) = \frac{\left\|[\dot\gamma, \ddot\gamma]\right\|}{\|\dot\gamma\|^3},\quad \varkappa(t) = \frac{\det(\dot\gamma, \ddot\gamma, \dddot\gamma)}{\left\|[\dot\gamma, \ddot\gamma]\right\|^2}.$$

Для этих формул можно найти соответствующие дискретные аналоги. Кроме того, имеется много различных формул для "кривизны" и "кручения" ломаной, которые выдерживают предельный переход: если в бирегулярную кривую начать вписывать ломаные, все более измельчая длины их ребер, и для таких ломаных вычислять по этим формулам кривизну и кручение, то в пределе получатся кривизна и кручение самой кривой. В свое время авторы опробовали этот подход, протестировав целый ряд функций, однако соответствующие графики качественно не отличались от приведенных нами выше графиков для α- и β-углов (мы рассматривали ломаные с ребрами почти постоянной длины).

Эксплуатация формул Френе при изучении конформации полипептидов является достаточно популярной деятельностью, см. например [1], [2], [3].

Однако серьезное применение математических методов для выявления реальных закономерностей, регулирующих конформацию полипептидов, возможно лишь после выяснения того, какая часть имеющейся базы данных (PDB) описывает реальные феномены, а не фантомы, возникшие в результате тех или иных ошибок.

Список литературы

[1] Glasel J.A. and Deutscher M.P. (ed.), *Introduction to Biophysical Methods for Protein and Nucleic Acid Research*. Academic Press, London, N.-Y., 1995.

[2] Goriely A., Hausrath A., and Neukirch S. *The differential geometry of proteins and its applications to structure determination*. Biophysical Reviews and Letters, 2008, **3** (1–2), 77–101.

[3] Hu S., Lundgren M., and Niemi A.J. *The Discrete Frenet Frame, Infection Point Solitons And Curve Visualization with Applications to Folded Proteins*. arXiv:1102.5658v1, 2011.

[4] Pauling L., Corey R.B., and Branson H.R. *The structure of proteins; two hydrogen-bonded helical configurations of the polypeptide chain*. Proc. Natl. Acad. Sci., USA, 1951, **37**, 205–211.

[5] Боженко В.К., Иванов А.О., Мищенко А.С., Тужилин А.А., Шишкин А.М. *Геометрические модели в биологии: как и что можно моделировать*. Электронный журнал "Научная визуализация", 2009, **1** (1), 66—99.

[6] Балабаев Н.К., Шайтан К.В. *Компьютерное моделирование молекулярной динамики*. Компьютерное моделирование полимерных и биополимерных систем. Под ред. Иванова В.А., Рабиновича А.Л., Хохлова А.Р., М.: Книжный дом «ЛИБРОКОМ», 2009, 35–62.

[7] Шайтан К.В., Турлей Е.В., Голик Д.Н., Терёшкина К.Б., Левцова О.В., Федик И.В., Шайтан А.К., Ли А., Кирпичников М.П. *Динамический молекулярный дизайн био- и наноструктур*. Журнал российского химического общества, 2006, **50** (2), 53–65.

[8] Шайтан К.В. *Энергетическая поверхность и конформационная динамика молекул*. Электрохимия, 2003, **39** (2), 212–219.

[9] Шайтан К.В., Беляков А.А., Леонтьев К.М., Сарайкин С.С., Михайлюк М.Г., Егорова К.Б., Орлов М.В. *Геометрия энергетической поверхности и конформационная динамика: от углеводородов — к белкам и пептидам*. Хим. физ., 2003, **22** (2), 57–68.

[10] Иванов А.О., Мищенко А.С., Тужилин А.А. *Геометрия ломаных и полипептидов*. Наноструктуры. Математическая физика и моделирование, 2014, **10** (1), 39–47.

[11] Иванов А.О., Мищенко А.С., Тужилин А.А. *Геометрия аминокислот и полипептидов*. Наноструктуры. Математическая физика и моделирование, 2014, **10** (1), 49–76.