Краткое описание и инструкция по использованию программы RNAmodel

Оглавление

1. Общие сведения

Программа RNAmodel (версия 2.8.3) предназначена для моделирования РНК-овой регуляции у бактерий методом Монте-Карло. Описание модели, особенности ее реализации и примеры использования приведены в [1-5]. Данный документ описывает локальную версию программы, которую пользователь может загрузить для исполнения на собственном компьютере. Для содержательного ознакомления и тестирования имеется также серверная версия программы, доступная на сайте http://lab6.iitp.ru/rnamodel.

Локальная версия программы позволяет *для заданного относительного уровня концентрации регулирующей аминокислоты* получить (путем моделирования некоторого числа траекторий в пространстве состояний модели) оценку вероятности события преждевременной терминации транскрипции вследствие срыва РНК-полимеразы с комплекса ДНК-РНК. Эта вероятность прямо характеризует степень экспрессии рассматриваемого гена при указанной концентрации. Чтобы получить не одну точку, а целую зависимость степени экспрессии гена в диапазоне концентраций аминокислоты программу необходимо запускать для каждого интересующего значения. Для автоматизации этой работы удобно использовать командные сценарии (скрипты), примеры которых имеются в дистрибутивном комплекте.

В последующих разделах описываются: комплект поставки программы, порядок установки и запуска, входные данные и параметры командной строки, выходные данные и рекомендации по эффективному применению.

Описываемая версия программы (исполняемый модуль архитектуры x86) предназначена для моделирования на IBM-совместимом ПК с операционной системой Windows. Программа имеет интерфейс командной строки и рассчитана на запуск в среде командного процессора операционной системы. Язык программирования – C, компилятор Microsoft Visual Studio 2008 SP1, имя исполняемого модуля – rnamodel.exe. Целевой процессор – Intel 32-битной архитектуры. Целевые операционные системы – Microsoft Windows XP SP3, 7, Microsoft Windows Server 2003 SP2, 2008 R2. Использование программы RNAmodel на других типах процессоров и операционных систем возможно, но может потребовать дополнительного тестирования и/или перекомпиляции.

Руководитель проекта: д.ф.-м.н., проф. В.А. Любецкий, зав. лабораторией ИППИ РАН им. А.А. Харкевича. (http://lab6.iitp.ru/ru/contacts.html)

Разработчик программы: к.т.н. Л.И. Рубанов, в.н.с. ИППИ РАН им. А.А. Харкевича

Email: rubanov@iitp.ru

2. Комплект поставки программы

Список файлов, содержащихся в комплекте поставки, с их кратким описанием приведен в табл. 1.

Таблица 1

Имя файла	Описание
ReadmeXXX.doc	Настоящий документ (XXX - номер версии программы)
QuickStart.doc	Краткое руководство на английском языке
rnamodel.exe	Исполняемый модуль программы RNAmodel
vcredist_x86.exe	Установочный файл свободно распространяемых динамических библиотек Microsoft Visual Studio 2008 SP1. Если этот продукт не установлен на ПК, данный файл необходимо один раз выполнить.
EcE_trpE Sden_leuA	Примеры входных файлов для программы: регуляторная область оперона *trpE* у *Escherichia coli* и оперона *leuA* у *Shewanella denitrificans.*
table.bat	Файл сценария для запуска программы RNAmodel в диапазоне концентраций от 0 до 1 с шагом 0,05. Первым аргументом должно быть имя входного файла. Этот сценарий применяется при первом запуске программы для данной последовательности.
table2.bat	Файл сценария для запуска программы RNAmodel в диапазоне концентраций от 0 до 1 с шагом 0,05. Этот сценарий удобно применять при последующих запусках программы для данной последовательности, чтобы не перезаписывались ранее полученные результаты. Первым аргументом должно быть имя входного файла, вторым – уникальный идентификатор данного эксперимента, например, _1, _2 и т.д.
test.bat	Файл сценария для запуска программы RNAmodel на примерах, с использованием сценариев table.bat, table2.bat.
*.log	Выходные файлы протокола, полученные в результате работы программы RNAmodel в примерах с многократным запуском модели.
*.txt	Обработанные файлы протокола для импорта в Excel.
*.out	Выходные файлы, полученные в примерах одиночного запуска
*.html	Файлы траектории, полученные в примерах одиночного запуска
traject.css	Стилевой файл для просмотра файлов траектории в браузере

3. Установка и запуск программы

1. На ПК, где предполагается работа с программой RNAmodel, создать папку для использования программы (например, d:\rnamodel\) и скопировать в нее все файлы из комплекта поставки.
2. Если на данном ПК не установлен продукт Microsoft Visual Studio 2008 SP1, запустить файл vcredist_x86.exe и ответить на задаваемые вопросы программы.
3. Запустить командный процессор Windows и перейти в папку программы. Для удобства выполнения этой операции рекомендуется создать на рабочем столе ярлык для запуска командного процессора, в свойствах которого указать соответствующую рабочую папку.
4. Запустить команду `rnamodel`. В окно командного процессора должен быть выдан текст подсказки о параметрах программы.
5. Запустить команду `test`, чтобы выполнить сценарий test.bat для проверки работы

программы на данном ПК. Поскольку при этом имеющиеся файлы результатов будут перезаписаны, рекомендуется создать их копии в другом месте. В зависимости от быстродействия ПК, обработка всего сценария может занять порядка 3-5 минут.

6. По окончании обработки сопоставить полученные файлы результатов (*.html, *.log, *.out, *.txt) с эталонными копиями. Точного совпадения может не быть в силу различного быстродействия и случайных факторов, но результаты должны быть сопоставимы.
7. После этого, опираясь на приведенные примеры, можно переходить к подготовке исходных данных и сценариев для реальных расчетов.

4. Описание входных данных

Входные данные программы передаются через аргументы командной строки, которая должна иметь следующий общий вид (элементы в квадратных скобках необязательны):

```
rnamodel [параметры] infile [outfile]
```

- параметры позволяют изменять установленные по умолчанию режим работы программы и характеристики алгоритма; признаком параметра является предшествующий ему символ «дефис» (-) или «наклонная черта» (/);
- аргумент `infile` указывает имя текстового файла с исходной нуклеотидной последовательностью; может быть также указан абсолютный или относительный путь к файлу, в противном случае используется текущая папка;
- аргумент `outfile` указывает имя файла для записи результатов работы программы; если этот аргумент опущен, то будет использоваться имя входного файла с добавлением расширения ".out". Как и в случае входного файла, может быть указан абсолютный или относительный путь к файлу, в противном случае используется текущая папка.

В соответствии с соглашениями командного процессора Windows, заглавные и строчные буквы в командной строке интерпретируются одинаково. Если имя файла или путь к нему содержат пробелы, соответствующий аргумент должен указываться в кавычках.

Форматы входных данных программы RNAmodel приведены в разделах 5 и 6. Краткую подсказку о формате входных данных можно получить по команде `rnamodel` с параметром `-h` или `rnamodel -?`, или вообще без параметров (эквивалентные формы задания параметров: `/h`, `/?`). Описание выходных данных программы приведено в разделе 7.

5. Параметры командной строки

Программа RNAmodel воспринимает следующие параметры (перечислены в алфавитном порядке, заглавные и строчные буквы интерпретируются одинаково):

`-a` Если указан этот параметр, выходной файл открывается в режиме дозаписи в конец, что может быть полезно при запуске в пакетном режиме; по умолчанию файл перезаписывается. В частности, этот параметр используется в подготовленных скриптах table.bat, table2.bat, чтобы создать сводный протокол для всех значений концентрации.

`-al`<число> Этот параметр позволяет ввести поправку для энергии связи микросостояния по формуле (26) в [4], задавая значение α. По умолчанию α = 0, т.е. поправка не применяется.

Примечание: Здесь и далее вместо <число> записывается желаемое численное значение (без пробела после параметра). Если значение отрицательное, перед ним, т.е. сразу после ключа параметра, ставится знак минус. Если значение дробное, то разделителем целой и дробной части должна быть точка. Нулевую целую часть разрешается опускать, например, `-al.35` эквивалентно `-al0.35`. При записи числа

в показательной форме основание 10 указывается латинской буквой E или e, например, `-al2.5e-3` эквивалентно `-al0.0025`.

`-b`<число> Добавочный множитель β в уравнении (16) в [4]. По умолчанию равен 1, т.е. не используется. (Допускается указывать значение β=0, при этом уравнение не решается, а принимается $p = \pi/2h$).

`-c`<число> Относительная величина концентрации c регулирующей аминокислоты (см. формулу (24) в [4]). Указывается число в интервале [0, 1]; значение по умолчанию 0,5.

`-cz`<число> Эталонное значение c_0, относительно которого задается концентрация (см. формулу (24) в [4]). Значение по умолчанию 1.

`-d`<xxx> Параметр указывает название регулирующей аминокислоты. Вместо <xxx> стоит одно из 20 общепринятых трехсимвольных сокращений (регистр букв не имеет значения): ALA – аланин, ARG – аргинин, ASN – аспарагин, ASP – аспарагиновая кислота, CYS – цистеин, GLN – глутамин, GLU – глутаминовая кислота, GLY – глицин, HIS – гистидин, ILE – изолейцин, LEU – лейцин, LYS – лизин, MET – метионин, PHE – фенилаланин, PRO – пролин, SER – серин, TRP – триптофан, THR – треонин, TYR – тирозин, VAL – валин. По умолчанию это триптофан, т.е. `-dtrp`.

`-d2`<xxx>, `-d3`<xxx> Эти параметры аналогичны параметру `-d` и позволяют указать вторую (и третью) аминокислоту в случае разветвленных аминокислот. Например, для *ilv*-оперона указываются параметры `-dile -d2leu -d3val`. Значения по умолчанию нет, т.е. если эти параметры не указаны, учитывается только одна аминокислота, заданная параметром `-d`.

`-e`<число> Указывает вариант используемой формулы для скорости переходов между макросостояниями. Параметр интерпретируется следующим образом:

`-e0`: асимметричная (формулы (4, 5) в [4]);
`-e1`: симметричная (формула (6) в [4]);
`-e2`: асимметричная формула с поправкой из раздела 4.1(5-c) [4];
`-e3`: симметричная формула с поправкой из раздела 4.1(5-c) [4].

По умолчанию принимается вариант 1.

`-fa` Задействует упрощенный вариант формулы (14) в случае шпильки, составленной из нескольких спиралей (по умолчанию не используется).

`-fd`<число> Задает значение параметра δ в формуле (14) в [4], по умолчанию 30.

`-fl`<число> Задает значение параметра L_2 в формуле (14) в [4], по умолчанию 27,1.

`-fm`<число> Выбор в алгоритме способа учета общего замедления РНК-полимеразы от набора шпилек вторичной структуры:

`-fm0`: учет только одной шпильки, максимально действующей на РНК-полимеразу, согласно формуле (21) в [4];

`-fm1`: учет суммарного замедления от всех шпилек вторичной структуры согласно формуле (20) в [4].

По умолчанию принимается первый вариант (только наиболее сильная шпилька).

`-fp`<число> Задает значение параметра p_0 в формуле (14) в [4], по умолчанию 0,1826.

`-fr`<число> Задает значение параметра r_0 в формуле (14) в [4], по умолчанию 1,0.

`-glb`<число> Задает значение параметра B в формуле (2) в [4] для петли шпильки, по умолчанию 6,5.

`-glc`<число> Задает значение параметра *C* в формуле (2) в [4], по умолчанию 5.

`-gli`<число> Задает значение параметра *B* в формуле (2) в [4] для внутренней петли (иначе – двустороннего выпячивания), по умолчанию 0.

`-glp`<число> Задает значение параметра *B* в формуле (2) в [4] для выпячивания, по умолчанию 4.

`-h` Вывод подсказки о формате командной строки и параметрах программы.

`-h`<число> Минимально допустимая длина плеча спирали (нт), по умолчанию 3.

`-hh`<число> Минимально допустимая длина плеча гипоспирали (нт), по умолчанию 3.

`-i`<число> Максимально допустимое число переходов между макросостояниями при фиксированном окне. По достижении этого порога производится принудительное изменение окна, т.е. сдвиг рибосомы или полимеразы, либо срыв полимеразы (исход разыгрывается обычным способом, но без учета вариантов, связанных с переходом в новое макросостояние). Значение по умолчанию 50000.

`-j`<число> Режим по умолчанию эквивалентен `-j0`. Если указать положительное число, алгоритм моделирования запускается в особом режиме (с неподвижными рибосомой и РНК-полимеразой) с целью нахождения установившегося состояния вторичной структуры РНК. При этом заданное число интерпретируется как максимальное число таких отыскиваемых равновесных состояний.

`-jf`<число> Используется только в режиме поиска равновесных состояний (см. выше параметр `-j`). Определяет критерий равновесия: если указано `-jf0`, выбираются состояния с минимальной энергией; если указано `-jf1`, выбираются более часто возникающие состояния. Значение по умолчанию 0.

`-ji`<число> Используется только в режиме поиска равновесных состояний (см. выше параметр `-j`). Позволяет исключить из рассмотрения начальный участок траектории (до выхода в окрестность равновесного состояния). Значение указывает число анализируемых макросостояний от конца траектории. Значение по умолчанию 0 интерпретируется как анализ всех макросостояний без изъятия.

`-k`<число> Задает параметр замыкания κ, значение которого соответствует представлению о «вязкости» цитоплазмы. По умолчанию принимается κ = 1000.

`-kn`<число> Режим по умолчанию эквивалентен `-kn0`, при этом допускаются только вторичные структуры РНК, не содержащие псевдоузлов. Если указано `-kn1`, то допускается образование псевдоузлов, и их энтропия вычисляется, как описано в [6]. При указании ненулевого значения, отличного от 1, вычисленная энтропия дополнительно умножается на это значение.

`-lapo`<число> Константа скорости перехода полимеразы на следующий нуклеотид (при отсутствии замедления под влиянием вторичной структуры в окне). По умолчанию 40.

`-lari`<число> Константа скорости перехода рибосомы на следующий кодон (на нерегуляторных кодонах). По умолчанию 15.

`-laur`<число> Константа скорости срыва РНК-полимеразы с нуклеотидов, находящихся на U-богатом участке. По умолчанию 10.

`-lmin`<число> Минимальная длина петли спирали (нт), по умолчанию 3.

`-lmax`<число> Максимальная длина петли спирали (нт), по умолчанию 50.

`-lpol`<число> «Размер» РНК-полимеразы от места выхода цепи РНК до точки

транскрипции (нт), по умолчанию 5.

`-lrib`<число> «Размер» рибосомы от ее P-участка до 3'-конца (нт), по умолчанию 12.

`-lura`<число> Минимально допустимая длина U-богатого участка (иначе полиурацила) для 1-го способа его определения (нт), по умолчанию 5. См. также параметр `-u`. (Подробнее о способах определения полиурацила см. раздел 8).

`-me`<число> Вносимая вручную поправка к величине энергии спирали, заданной параметром `-mn`. Указанное число вычитается из вычисленной обычным образом величины энергии. Используется для экспериментов. Значение по умолчанию 0.

`-mh`<число> Минимальная длина плеча гипоспирали, начиная с которой вносится поправка, указанная параметром `-me`, по умолчанию 5.

`-ml`<число> Значение параметра $l_{\max}$ в формуле (26) в [4], по умолчанию 10.

`-mn`<число> Номер спирали, энергия которой корректируется с помощью параметров `-me`, `-mh`. Конкретно, если текущее микросостояние содержит гипоспираль спирали с указанным номером, и длина плеча гипоспирали не менее указанной параметром `-mh`, то из энергии этой гипоспирали дополнительно вычитается поправка, заданная параметром `-me`. Номера спиралей для каждой последовательности назначаются алгоритмом; список спиралей можно получить с помощью параметра `-o2`. По умолчанию принят фиктивный номер 0, что означает отсутствие поправок.

`-n`<число> Вариант используемого генератора псевдослучайной последовательности:

- `-n0`: «минимальный» случайный датчик Park-Miller (период равен 2^{31}–2);
- `-n1`: датчик Park-Miller с перетасовкой Bays-Durham (для числа вызовов до 10^8);
- `-n2`: датчик L'Ecuyer с перетасовкой Bays-Durham (период более $2\cdot10^{18}$);
- `-n3`: датчик Кнута (не мультипликативный);
- `-n4`: датчик ANSI (реализация функции *rand*( ) языка C).

По умолчанию используется вариант `-n1`. Для последовательностей длиной 200 нт и более рекомендуется использовать вариант `-n2`. Вариант `-n3` может быть полезен в случае сомнений в статистической достоверности результатов. Варианты `-n0` и `-n4` быстрые, но удовлетворяют не всем статистическим тестам, к тому же последний зависит от реализации конкретного компилятора.

`-o`<число> Параметр задает режим выдачи информации в выходной файл протокола (см. раздел 7). Указанное значение интерпретируется следующим образом:

- `-o0`: минимальная выдача – исход моделирования для каждой траектории;
- `-o1`: последовательность сдвигов рибосомы и РНК-полимеразы для каждой траектории;
- `-o2`: выдать полный список спиралей для данной последовательности;
- `-o4`: выдавать номера спиралей, попадающих в каждое окно (T-список);
- `-o8`: выдавать список всех макросостояний для текущего окна;
- `-o16`: выдавать список всех микросостояний для текущего окна;
- `-o32`: отображать каждый переход модели вдоль траектории.

Значения 2...12 могут комбинироваться, для чего достаточно указать их сумму. По умолчанию используется режим `-o1`. При проведении массовых расчетов рекомендуется указывать режим `-o0`.

`-oh`<число> По умолчанию используется режим `-oh0`; при указании ненулевого значения будет создан выходной файл траектории в формате HTML (см. раздел 7). Указанное число интерпретируется как сумма следующих значений:

-oh1: выдавать строку с величинами энергии, «силы» торможения полимеразы (на U-богатых участках) и диаграммой текущего макросостояния;
-oh2: выдавать строки со всеми микросостояниями текущего макросостояния;
-oh4: не показывать в выдаче циклы любой длины (несовместимо с -oh8);
-oh8: показывать только макросостояния, в которых впервые образуется гипоспираль ранее не встречавшейся спирали (несовместимо с -oh4);
-oh16: показывать только одно микросостояние каждого макросостояния (если включен бит -oh2);
-oh32: показывать только финальное макросостояние траектории.

-oi<число> Параметр используется только в режиме -o2. Если указано -oi1, в файл протокола вместе со списком спиралей выдается обратный индекс принадлежности нуклеотидов спиралям. Значение по умолчанию -oi0 (индекс не выдается).

-ol<число> Параметр позволяет ограничить общее число строк, выдаваемых в файл траектории (см. параметр-oh). Выдается указанное число строк, непосредственно предшествующих концу траектории. Значение 0 интерпретируется как отсутствие лимита. Значение по умолчанию 2000.

-or<число> Параметр используется только в режиме -o2 и позволяет выдавать вместе со списком спиралей и матрицы отношений между спиралями [5]. Возможны варианты:

-or0: не выдавать матрицы отношений (режим по умолчанию);
-or1: выдавать матрицы в разреженном формате (без нулевых элементов);
-or2: выдавать матрицы в полном формате.

-pr<число> Максимальный размер l'' односторонних выпячиваний в черенке шпильки, которые считаются «малыми» при расчете замедления полимеразы; см. формулу (15) в [4]. Значение по умолчанию 2.

-ps<число> Максимальный размер l'' двусторонних выпячиваний (внутренних петель) в черенке шпильки, которые считаются «малыми» при расчете замедления полимеразы; см. формулу (15) в [4]. Значение по умолчанию 2.

-q<число> Задает параметр Q в формуле для поправки из раздела 4.1(5-c) в [4] к вычислению скорости переходов. См. также описание параметра -e. Значение по умолчанию 100.

-r<число> Задает упорядоченность спиралей при выводе их списка (если указан параметр -o2). По умолчанию принят режим -r0, при котором спирали упорядочены сначала по возрастанию B, потом по возрастанию C (A, B, C, D – концы плеч спирали при обходе цепи в направлении от 5' к 3'). Если указано ненулевое число, оно интерпретируется как сумма следующих значений (приоритет имеет большее):

1: упорядочить по возрастанию A, B, C, D;
2: упорядочить по возрастанию длины;
4: упорядочить по возрастанию энергии.

Примечание: Этот параметр может использоваться только для получения списка спиралей; при расчетах необходимо сохранять упорядоченность по умолчанию.

-sp<число> Задает положение полимеразы в начале моделирования, т.е. номер позиции нуклеотида последовательности, после которого начинается 5'-край полимеразы. Значение по умолчанию 13.

-sr<число> Задает положение рибосомы в начале моделирования, т.е. номер позиции нуклеотида последовательности, с которым связан P-участок рибосомы. Значение по умолчанию 1.

`-sv`<число> Параметр позволяет задать начальное условие для инициализации генератора псевдослучайной последовательности. Если параметр не указан, то генератор инициализируется часами компьютера (соответствующее значение `-sv` содержится в протоколе работы программы). Использование этого параметра позволяет получать воспроизводимые результаты, например, при отладке.

`-t`<число> Параметр позволяет указать термодинамическую температуру среды в градусах. Значение по умолчанию 310 (соответствует приблизительно 37°C).

`-u`<число> Этот параметр позволяет выбирать способ определения U-богатого участка (полиурацила) и интерпретируется следующим образом:

- если указано значение меньше 1, то используется 1-й способ, и это значение есть минимально допустимая доля букв U/T, чтобы считать некоторый участок последовательности U-богатым. Минимальная длина участка задается параметром `-lura`. Алгоритм анализирует участки любой длины не менее указанной, начиная с каждой позиции последовательности, и выделенные таким образом U-богатые участки объединяются в связные компоненты.
- если указано значение, большее или равное 1, то используется 2-й способ определения U-богатого участка, и это значение есть минимально требуемое число букв U/T в U-богатом участке. При этом параметр `-lura` не учитывается (т.е. длина участка не лимитируется), зато параметром `-ug` ограничивается сверху длина промежутка из других букв между буквами U/T внутри U-богатого участка. Алгоритм выделяет в последовательности все такие U-богатые участки, добавляя к каждому из них с обеих сторон нуклеотиды в количестве, равном половине допустимой длины промежутка.

Независимо от способа определения, имеется возможность ограничить рассмотрение только частью U-богатых участков, расположенных ближе к 3'-концу последовательности. Подробнее о способах определения полиурацила см. раздел 8. По умолчанию используется значение `-u.8` (т.е. применяется 1-й способ), и вместе со значением по умолчанию `-lura5` это означает, что минимально допустимый U-богатый участок должен содержать четыре буквы U/T из пяти.

`-ug`<число> Этот параметр используется только при 2-м способе определения U-богатого участка, указывая максимальную длину промежутка из других (не U/T) букв внутри U-богатого участка. См. также параметр `-u`. Значение по умолчанию 3.

`-un`<число> Число учитываемых U-богатых участков (связных компонент), считая от 3'-конца последовательности. См. также параметр `-u`. Значение по умолчанию 1, т.е. учитывается только самый последний U-богатый участок, а прочие игнорируются.

`-x`<список> Этот параметр позволяет указать участки последовательности, которые не должны участвовать в комплементарной связи (например, из-за связи с лигандом). Каждый участок задается парой чисел через запятую. Первое число – номер позиции начала участка во входной последовательности (начиная с 1), второе число – длина участка (нт). Всего можно указать до 6 участков, перечисляя их через запятую в произвольном порядке. Поле не должно содержать пробелы или другие символы, помимо цифр и запятых. По умолчанию таких исключений нет.

`-z`<число> Задает число траекторий моделирования, на котором оценивается значение вероятности события преждевременной терминации. Чем больше берется траекторий, тем достовернее оценка (но и дольше длится моделирование).

Примеры использования параметров программы содержатся в файлах сценариев, входящих в дистрибутивный комплект, а также в разделе 8 данного описания.

6. Входной файл

Имя файла исходных данных должно быть указано аргументом `infile` в командной строке запуска программы. Этот текстовый файл должен содержать в первой строке исходную нуклеотидную последовательность. Последовательность состоит из строчных или прописных алфавита {A, C, T, G, U}, в котором буквы T и U считаются эквивалентными, другие символы не допускаются (выдается сообщение об ошибке). Символ конца строки не обязателен, последующие строки файла игнорируются.

Примеры входных файлов для программы RNAmodel имеются в дистрибутивном комплекте. Рекомендуется давать файлам имена без расширений, хотя это не обязательно (по умолчанию в таком случае выходные файлы будут иметь более одного расширения).

7. Описание выходных данные

7.1 Код возврата

Нулевой код возврата программы (возвращаемое значение головной функции) является признаком успешного окончания моделирования. Любые другие значения являются результатом ошибок при запуске (параметры командной строки) или исполнении программы, о чем также выдаются сообщения на консоль и в файл протокола. Проверка кода возврата полезна при составлении командных файлов (скриптов) для пакетной обработки.

7.2 Выходной файл протокола

Содержимое выходного файла протокола зависит от указанных параметров программы. В общем случае он содержит заголовочную часть, данные по каждой траектории и финальную часть. Ниже они описываются по порядку, причем в описании используются примеры файлов протокола из дистрибутивного комплекта. (Заметим, что с помощью параметра `-a` протоколы серии запусков можно объединять в один; описание ниже игнорирует этот факт и относится к одиночному протоколу, хотя примеры *.log из дистрибутивного комплекта являются именно такими объединенными протоколами).

7.2.1 Заголовочная часть протокола

Как минимум, содержит 4 строки, в которых воспроизведены значения параметров, использованных при запуске программы. Прокомментируем их на примере файла Sden_leuA_1.log (табл. 2):

Таблица 2. Шапка протокола

Строка	Элемент	Описание или соответствующий параметр запуска
2	`Sden_leuA`	Имя входного файла (аргумент `infile` командной строки)
	`c=0`	`-c0` (эквивалентное значение параметра)
	`c0=1`	`-cz1` (по умолчанию)
	`aacid=LEU`	`-dLeu` (здесь параметры `-d2`, `-d3` не использовались, в противном случае наименования нескольких аминокислот приводятся через запятую)
	`L1=27.10`	`-fl27.1` (по умолчанию)
	`p0=0.1826`	`-fp.1826` (по умолчанию)
	`r0=1.000`	`-fr1` (по умолчанию)
	`delta=30.00`	`-fd30` (по умолчанию)
	`beta=1`	`-b1` (по умолчанию)
	`MAX`	`-fm0` (по умолчанию). Если задан параметр `-fm1`, то вместо `MAX` будет указано `SUM`.

Строка	Элемент	Описание или соответствующий параметр запуска
3	alpha=0	-al0 (по умолчанию)
	Bl=6.50	-glb6.5 (по умолчанию)
	Bp=4.00	-glp4 (по умолчанию)
	Bi=0.00	-gli0 (по умолчанию)
	C=5.00	-glc5 (по умолчанию)
	H=3	-h3 (по умолчанию)
	HH=3	-hh3 (по умолчанию)
	K=1.00e+003	-k1e3 (по умолчанию)
	ex={ }	Если используется параметр -x, внутри фигурных скобок воспроизводится заданный этим параметром список исключаемых участков последовательности
	нет	Если с помощью параметров -mn, -me, -mh в энергию какой-то спирали вносится поправка, будет указан элемент вида man={153 5.0 5}, где в фигурных скобках указаны значения этих параметров в соответственном порядке
4	minloop=3	-lmin3 (по умолчанию)
	maxloop=50	-lmax50 (по умолчанию)
	ribolen=12	-lrib12 (по умолчанию)
	polylen=5	-lpol5 (по умолчанию)
	uralen=5(f>0.800,#1)	Параметры полиурацила при 1-м способе его поиска: -lura5 (по умолчанию) -u.8 (по умолчанию) -un1 (по умолчанию). Если использован 2-й способ задания U-богатого участка, этот элемент будет иметь типовой вид U-run=3(g=3,#1), что соответствует набору параметров -u3 -ug3 -un1.
	T=310	-t310 (по умолчанию)
	ef=1	-e1 (по умолчанию)
	knot=0	-kn0 (по умолчанию)
	v.2.8.3	Номер версии программы RNAmodel.
5	rib=15.0	-lari15 (по умолчанию)
	pol=40.0	-lapo40 (по умолчанию)
	ur=10.0	-laur10 (по умолчанию)
	maxbulge=(2,2)	-pr2 -ps2 (оба по умолчанию)
	start=(13,13)	В скобках указаны номера позиции 5'- и 3'-концов окна в начале моделирования. Это соответствует значениям параметров -sr1 (при -lrib12) и -sp13 (оба по умолчанию)
	z=100	-z100 (по умолчанию)
	[777]	-sv777 Если параметр -sv не был задан, в скобках стоит значение, использованное для запуска датчика псевдослучайной последовательности.
	maxiter=50000	-i50000 (по умолчанию)
	нет	Если параметром -e было задано значение 2 или 3, здесь выдается элемент вида qf100.0, сообщающий значение параметра -q

Далее, если при запуске программы был указан параметр `-o2` (или другая сумма допустимых значений, содержащая слагаемое 2), в заголовочной части выдается список спиралей (см. пример в файле протокола EcE_trpE.log):

Первая строка списка (строка 7 протокола) начинается словом HELICES и содержит в правой части разметку позиций последовательности точками или символами U, если позиция принадлежит U-богатому участку. В последующих строках в этих позициях выводятся символы последовательности, а эта строка позволяет контролировать, как были выделены U-богатые участки при заданных параметрах запуска программы.

Последующие строки списка (для примера рассмотрим строку 12 протокола) перечисляют все непродолжаемые спирали, возможные в данной последовательности, в порядке, заданном значением параметра `-r`. Строка содержит следующие элементы:

5)	номер, присвоенный спирали (номера, которые появляются, когда спирали упорядочены по умолчанию, т.е. по возрастанию B, затем C, и должны задаваться в параметре `-mn`)
1 7 11 17	Позиции A, B, C, D концов плеч данной спирали в последовательности.
-7.700e+000	Энергия стекинга спирали (в ккал/моль) без учета стекинга концевых пар.
+6.282e+000	Свободная энергия петли спирали (в ккал/моль).
-1.418e+000	Сумма двух вышеприведенных значений (ориентировочное значение общей энергии спирали).
7	Длина плеча спирали (нт), равная B-A+1=D-C+1.

Далее воспроизведена исходная последовательность (с точностью до перекодировки U -> T), причем буквы в плечах спирали заглавные, а все остальные – строчные.

Если в дополнение к параметру `-o2` при запуске программы были указаны параметры `-oi` и/или `-or`, после списка спирали выдается обратный индекс принадлежности нуклеотидов спиралям и/или матрицы отношений между спиралями (не описывается).

7.2.2 Данные протокола для каждой траектории

Число траекторий моделирования задается при запуске программы параметром `-z`. По умолчанию исполняется только одна траектория, что, конечно же, не позволяет оценить вероятность терминации, но зато помогает оценить время моделирования для данной последовательности и правильно выбрать число траекторий.

Для каждой траектории выдаются одинаковые данные, состав которых зависит от значения параметра `-o`.

Если указано значение `-o0`, выдается только одна строка сводных данных о траектории; в качестве примера прокомментируем строку 40 файла Sden_leuA_1.log:

33:	порядковый номер траектории, начиная с 0
Anti-terminated	Исход моделирования вдоль этой траектории, здесь анти-терминация (в противном случае стоит Terminated)
x= 28	Позиция 5'-края окна в конце моделирования
(r)	Вид кодона, на котором находится P-участок рибосомы, в данном случае это регуляторный кодон одной из регулирующих аминокислот, указанных параметрами `-d`, `-d2`, `-d3` при запуске программы. Другие возможные варианты: (s) означает стоп-кодон, (x) – прочие кодоны.
y=129	Позиция 3'-края окна в конце моделирования

3100 iterations Общее число переходов вдоль траектории, т.е. смен макросостояния и/или движений рибосомы и полимеразы.

0 overruns Число раз, когда было превышено максимальное число смен макросостояния в одном и том же окне, заданное параметром `-i`.

0 knots Число раз, когда возникало макросостояние, содержащее псевдоузел.

0/2.7 s Первое число указывает компьютерное время моделирования, второе – физическое время (оба значения даны в секундах).

[0.118] Текущая оценка вероятности терминации по уже пройденным траекториям моделирования.

Если указано значение `-o1`, перед строкой сводных данных о траектории дополнительно выдаются более подробные данные о ходе траектории. В качестве примера рассмотрим выдачу в файле Sden_leuA.out. В строке 7 файла указан заголовок таблицы, выдаваемой в этом режиме, поэтому будем ссылаться при описании на имена соответствующих столбцов.

Каждая строка таблицы выдается при изменении краев текущего окна, т.е. сдвиге рибосомы на один кодон или полимеразы на один нуклеотид. Исключением является случай, когда сдвиг рибосомы оказывается невозможным из-за недостаточной ширины окна (т.е. рибосома «догоняет» полимеразу), и позиции краев окна не меняются. Содержимое строки:

X Текущая позиция 5'-края окна

Y Текущая позиция 3'-края окна

L Длина текущего окна (нт)

Tlst Длина Т-списка, т.е. число непродолжаемых спиралей из общего списка возможных, которые пересекаются с текущим окном хотя бы на длину минимальной гипоспирали. Напомним, что при моделировании переходов в фиксированном окне учитываются только спирали из Т-списка

Macro Число различных макросостояний, встретившихся при переходах в этом окне

Micro Число различных микросостояний, встретившихся при переходах в этом окне

Diag Максимальная длина диаграммы макросостояния (число его гипоспиралей)

Knot Максимальное число псевдоузлов макросостояния

Iter Общее число перемен макросостояния в этом окне

Peak Максимальное число повторений одного и того же макросостояния

Time Компьютерное время от начала моделирования данной траектории (с)

Tmod Физическое время от начала данной траектории (с)

Если указаны другие значения параметра `-o`, отличные от 0 и 1, то перед строкой сводных данных о траектории выдаются еще более подробные данные о ходе траектории, которые здесь не описываются. Объем выдаваемого протокола может достигать гигабайт даже для одной траектории, что позволяет проследить ход моделирования в мельчайших деталях.

7.2.3 Финальная часть протокола

Финальная часть протокола состоит из двух строк, примером которых являются строки 107–108 файла Sden_leuA_1.log. В первой строке финальной части протокола содержатся следующие данные:

Probability of termination: 0.120

Оценка вероятности терминации по индивидуальным исходам выполненных

траекторий моделирования.

Y=12 Число траекторий, завершившихся терминацией

N=88 Число траекторий, завершившихся анти-терминацией

O=0 Общее число раз, когда превышалось максимальное число перемен макросостояния в одном и том же окне, заданное параметром `-i`.

I=278509 Суммарное число переходов, т.е. смен макросостояния и/или движений рибосомы и полимеразы, вдоль всех траекторий моделирования.

R=100 Число траекторий, в конце которых P-участок рибосомы находился на регуляторном кодоне.

S=0 Число траекторий, в конце которых P-участок рибосомы находился на стоп-кодоне.

X=0 Число траекторий, в конце которых P-участок рибосомы находился на других кодонах.

Time=11/308.2 s Суммарное время моделирования всех траекторий (11 с) и суммарное физическое время этих траекторий (308 с).

Вторая заключительная строка содержит служебную информацию (не описывается).

Если программа RNAmodel была запущена с параметром -a, как, например, в составе сценария, аналогичного имеющимся в комплекте поставки, то формируется объединенный файл протокола из нескольких порций, аналогичных описанным выше. Иногда удобно выделить из всего такого протокола, имеющего большой объем, только сводку результатов каждого запуска программы. Такой прием использован в сценариях table.bat, table2.bat; он позволяет получить файл, состоящий только из строк финальной части протокола для каждого задаваемого значения концентрации аминокислоты. Этот файл затем легко импортируется в электронную таблицу, например, Excel, для консолидации результатов и/или автоматизированной обработки. (Примеры таких файлов с расширением *.txt содержатся в дистрибутивном комплекте).

7.3 Информация, выдаваемая на консоль

Протокол, выдаваемый на консоль оператора, позволяет следить за работой программы. В основном он аналогичен содержимому файла протокола, выдаваемого в режиме `-o0`. Следует иметь в виду, что при большом числе траекторий консольная выдача занимает заметную долю общего времени счета, поэтому для ускорения процесса можно полностью запретить выдачу на консоль стандартными средствами операционной системы (`>nul` в конце командной строки запуска). В таком случае за работой программы можно будет следить только путем просмотра текущего содержимого файла протокола с помощью внешней программы.

7.4 Файл траектории

В дополнение к выходному файлу протокола, имеется возможность наглядно отображать ход моделирования вдоль одной или нескольких траекторий посредством выходного файла траектории в формате HTML. В совокупности с подготовленным файлом стилей, этот файл траектории позволяет просматривать последовательность возникающих макросостояний вторичной структуры РНК с помощью обычного Web-браузера (проверено использование Internet Explorer с версии 6 и Google Chrome с версии 24). В основном, файл траектории предназначен для того, чтобы упростить анализ динамики изменения вторичной структуры в конкретной ситуации, когда исход моделирования отличается от ожидаемого. Благодаря наглядности представления и другим удобным возможностям, таким как исключение из демонстрируемой траектории многократно повторяющихся циклов перехода между одной и

той же последовательностью макросостояний, анализ значительно упрощается по сравнению с использованием текстового файла протокола.

Для получения файла траектории при запуске программы RNAmodel необходимо указать ненулевое значение параметра `-oh`. В этом параметре бит 0 (имеющий значение 1) включает показ строк с диаграммой каждого макросостояния в форме последовательностей плеч гипоспиралей с их номерами, а также значения энергии макросостояния и действующей силы *F*; бит 1 (со значением 2) включает показ каждого микросостояния этого макросостояния прямо на исходной последовательности в виде отдельной строки; бит 2 (со значением 4) включает вышеупомянутый режим, при котором любой цикл перехода между группой макросостояний показывается только один раз. Задавая соответствующее значение параметра `-oh`, можно установить любую желаемую комбинацию битов. В частности, содержащийся в дистрибутивном комплекте файл траектории EcE_trpE.html был получен при значении параметра `-oh2`, а файл Sden_leuA.html – при значении параметра `-oh7`.

Файлу траектории всегда присваивается то же имя, что и файлу протокола, но с расширением .html. Содержимое файла траектории опишем на примере Sden_leuA.html. В начале файла выдается такие же заголовочная часть из четырех строк и строка с разметкой положения U-богатых участков, что и в файле протокола. Затем воспроизведена последовательность макросостояний вдоль траектории в следующем виде.

Если бит 0 в параметре `-oh` включен (как в этом случае), сначала выводится строка с диаграммой макросостояния, которая начинается с порядкового номера итерации. Например, рассмотрим строку после итерации 3100 (вблизи конца файла):

```
3100: G=-7.151 F=18.190 -178 -183 183 178
```

Здесь после номера итерации приведено средняя (по всем микросостояниям с учетом их априорной вероятности) энергия макросостояний. Если, как в данном случае, 3'-конец полимеразы находится на полиурациле, то указывается вычисленное значение силы замедления полимеразы, также усредненное с учетом априорной вероятности микросостояния. После этого стоит диаграмма макросостояния, которое в данном случае состоит из двух образующих шпильку гипоспиралей, принадлежащих спиралям с номерами 178 и 183, в соответствии с нумерацией спиралей в списке, выдаваемом при указании параметра `-o2`. Элементы диаграммы приведены в порядке обхода последовательности от 5' к 3', отрицательный номер отвечает левому плечу гипоспирали, положительный – правому. Могут встречаться макросостояния значительно более сложной структуры, в том числе с псевдоузлами, если исходные данные и параметры запуска допускают образование псевдоузлов (в этих случаях после диаграммы стоит слово Pseudoknot).

Если бит 1 в параметре `-oh` включен (как в этом случае), после строки с диаграммой приведена одна или более строк, по числу микросостояний данного макросостояния (здесь таких строк две). В каждой строке воспроизведена исходная последовательность с цветовой разметкой текущего окна и гипоспиралей микросостояния. А именно, буквы нуклеотидов последовательности справа и слева от текущего окна показаны притушенным шрифтом, буквы внутри окна – нормальным шрифтом. Буквы плеч спиралей заглавные, и снабжены различным фоном для разных гипоспиралей (но одинаковым для обоих плеч), как и на диаграмме. В конце строки для первого (или единственного) микросостояния показано значение физического времени в секундах от начала моделирования.

Отметим, что в случае сдвига рибосомы или полимеразы указанный в файле номер итерации изменяется на 2, т.к. фактически показан результат после двух событий: изменение границ окна и корректировка макро-/микросостояний с учетом нового окна.

Если включен бит 2 в параметре `-oh`, то часть итераций может отсутствовать в файле траектории, так как они являются одним или несколькими повторениями той же последовательности нескольких состояний, что и выше (т.е. в этом режиме от любого цикла в ходе траектории в файл включается только один проход).

Полная траектория, даже при отбрасывании циклов, может занимать большое число строк. Поэтому при анализе статистики событий на материале нескольких траекторий рекомендуется выдавать только те макросостояния, в которых участвует хотя бы одна ранее не встречавшаяся спираль, для чего в этом параметре вместо бита 2 устанавливается бит 3 (со значением 8). При выводе микросостояний можно дополнительно уменьшить объем выдаваемой информации, устанавливая бит 4 (со значением 16), тогда для каждого макросостояния будет выдаваться только одно микросостояние, независимо от их числа. Наконец, минимальная выдача включает только финальное макросостояние каждой траектории, а исход моделирования указан справа от последовательности в виде T.. (терминация) или ..A (антитерминация). В этом режиме, например, с параметром `-oh59`, вполне удается вывести в один файл данные по 100 траекториям (-z100), тогда как при полном показе в режиме -oh7 даже 5-10 траекторий отображаются неприемлемо большим файлом, который очень долго (если вообще) открывается браузером.

8. Рекомендации по эффективному применению

8.1 Подготовка исходной последовательности

Решающее значение для успешного моделирования имеет правильная подготовка исходной последовательности, т.е. выделение нужного участка из всей лидерной области интересующего гена. Рекомендуется, чтобы исходная последовательность начиналась со старт-кодона гена лидерного пептида. Тем самым будут обеспечены правильная рамка считывания и начальные условия для параметров запуска по умолчанию. Можно вырезать последовательность и с началом раньше, но тогда параметром `-sr` следует указать соответствующую позицию начала трансляции. В наших экспериментах встречались лидерные области без старт-кодона, в таких случаях начало последовательности произвольно выбиралось до начала регуляторных кодонов, но так, чтобы для них и для стоп-кодона была установлена правильная рамка считывания.

Что касается конца исходной последовательности, то следует учитывать, что продолжительность моделирования изменений состояния вторичной структуры РНК при фиксированном окне между рибосомой и полимеразой быстро растет с увеличением длины окна, и при длине окна порядка 200 нт становится неприемлемо большой. С учетом того, что при малых значениях концентрации рибосома вплоть до конца моделирования остается вблизи первого регуляторного кодона, в качестве ориентира за максимальный размер окна можно принять расстояние от начала кодона, следующего за первым регуляторным кодоном, до конца последовательности минус сумма «размеров» полимеразы и рибосомы (они задаются параметрами `-lpol` и `-lrib` соответственно).

В общем, чем короче последовательность, тем быстрее она обрабатывается. Идеальное место, где можно обрезать 3'-конец последовательности, лежит справа от U-богатого участка, но не дальше 15-20 нт от 3'-конца терминатора, поскольку на таком расстоянии его действие по срыву полимеразы с комплекса ДНК/РНК практически сходит на нет. Здесь следует, конечно, учитывать длину полимеразы (параметр `-lpol`) и значение параметра r_0 (задается параметром `-fr` программы), а также используемый способ определения U-богатого участка и то, где он реально выделяется в данной последовательности.

Если местоположение терминатора и антитерминатора в данной последовательности неизвестно или неоднозначно, либо U-богатый участок слабо выражен или неочевиден, рекомендуется вначале запустить модель на необрезанной исходной последовательности с параметрами `-lmax0 -un10 -o2`, чтобы получить максимально полный список U-богатых участков, определяемых по умолчанию, и спиралей - кандидатов на роль антитерминатора и терминатора, после чего прервать работу программы, не дожидаясь конца моделирования. Анализируя полученный список, можно выделить альтернативные пары терминатор-антитерминатор и далее обрезать последовательность в соответствии со сделанным выбором, иногда – в нескольких вариантах. Наличие в каком-то варианте характерной зависимости вероятности терминации от концентрации аминокислоты может быть убедительным

аргументом в пользу наличия классической аттенюаторной регуляции (КАР) в этой регуляторной области, а также правильности выбора терминатора и антитерминатора.

8.2 Максимальный размер петли спирали

Ограничение максимальной длины петли позволяет уменьшить число рассматриваемых спиралей и, следовательно, ускорить моделирование за счет сокращения перебора. Сама по себе спираль с большой петлей характеризуется большим положительным значением энергии и поэтому ее появление маловероятно (такая спираль нестабильна). Однако здесь следует учитывать, что у длинной шпильки, например, антитерминаторной, черенок формально как раз является спиралью с длинной петлей, хотя значительная часть нуклеотидов этой петли может быть связана между собой, образуя другие спирали шпильки. Поэтому необходимо ограничивать длину петли так, чтобы этому условию формально удовлетворяли все гипоспирали черенка.

Установленное по умолчанию значение параметра `-lmax50` является компромиссным; в ряде примеров в наших экспериментах его приходилось увеличивать до 70 и даже до 100. Если вторичная структура известна, то следует проверить координаты плеч черенка антитерминатора и установить значение параметра `-lmax` не менее D–A–4. Следует иметь в виду, что при проверке длины спиралей алгоритм использует строгое неравенство, т.е. допускаются только спирали с длиной петли меньшей, чем указано параметром `-lmax`.

Если вторичная структура неизвестна, то рекомендуется вначале запустить модель без ограничения длины петли (см. п. 8.1), а потом уточнить ограничение на основе фактических характеристик кандидатов на роль черенков шпилек в этой последовательности.

8.3 Параметры выделения U-богатого участка

Детали механизма срыва РНК-полимеразы с цепи ДНК до конца не известны. Наша модель предполагает, что срыв может произойти только в том случае, когда точка транскрипции находится в пределах U-богатого участка, и при условии, что вблизи имеется шпилька вторичной структуры, оказывающая достаточное для срыва действие на полимеразу.

Таким образом, наличие U-богатого участка вблизи терминатора оказывается в нашей модели необходимым (но, конечно, не достаточным) условием КАР. Если оно не выполнено, то результат моделирования заведомо окажется отрицательным, поэтому перед запуском модели на конкретной последовательности необходимо убедиться в наличии такого участка.

Для этого понятия не существует общепринятого определения. В модели реализованы два способа выделения U-богатых участков, каждый использует свой набор параметров. Выбор подходящего для данной последовательности способа и значений его параметров остается за пользователем программы.

Первый способ. В этом способе используется следующее неявное определение U-богатого участка. Если некоторый участок последовательности, имеющий длину $l \geq l_{\min}$, содержит не менее $f_{\min} \cdot l$ букв U (или T), то все нуклеотиды этого участка считаются принадлежащими U-богатому участку. Минимальная длина участка $l_{\min}$ и относительная доля $f_{\min}$ букв в нем являются параметрами, значения которых задаются при запуске программы параметрами, соответственно, `-lura` и `-u` (см. раздел 5). По умолчанию $l_{\min} = 5$, $f_{\min} = 0{,}8$.

В результате использования такого определения все позиции последовательности разбиваются на два класса: принадлежащие и не принадлежащие U-богатому участку. Каждая из полученных связных компонент и является U-богатым участком, на котором теоретически возможен срыв полимеразы в нашей модели. Опыт использования модели показал, что наличие таких участков левее петли терминатора часто искажает картину регуляции, поэтому в программе предусмотрена возможность не учитывать их, а ограничиться только заданным числом U-богатых участков, считая от конца последовательности (по умолчанию – только одним, самым последним). Для этого

используется параметр `-un` (см. раздел 5). При обрезании конца последовательности необходимо учитывать значение этого параметра (изменяя его, если это требуется), чтобы гарантированно выделялся U-богатый участок, ближайший к терминатору, на котором ожидается срыв полимеразы в случае терминации.

Второй способ. При этом способе U-богатый участок определяется явно, а именно: это максимально продолженный участок последовательности, содержащий не менее n_{min} букв U (или T), так что промежутки между двумя соседними буквами U состоят из не более чем g_{max} других букв. На обоих концах U-богатого участка в него дополнительно будем включать еще g_{max} /2 соседних букв. Значения n_{min} и g_{max} задаются при запуске программы, соответственно, параметрами `-u` и `-ug` (см. раздел 5). Таким образом, с помощью параметра `-u` осуществляется одновременно и выбор одного из обсуждаемых двух способов, поскольку при первом способе значение этого параметра не больше единицы, а при втором способе – больше 1. По умолчанию максимальная длина промежутка g_{max} = 3, и рекомендуется задавать $n_{min} \geq 3$.

При использовании второго способа тоже может выделяться более одного U-богатого участка, и их число также можно ограничивать параметром `-un`, как и в первом случае.

Трудно дать универсальные рекомендации, в каком случае лучше применять каждый их способов. В общем, если U-богатый участок ближе к классическому, т.е. представлен в последовательности многочисленными буквами U с небольшими промежутками, то более адекватен первый способ, если букв U мало и промежутки между ними большие, то второй. В любом случае мы рекомендуем для выбранного набора параметров полиурацила один раз запустить программу с параметром `-o2` или `-oh2`, чтобы убедиться в правильном выделении U-богатого участка (напомним, что соответствующая разметка показана в начале списка спиралей и в конце заголовка в html-файле траектории).

Еще один важный аспект выделения полиурацила состоит в том, что U-богатый участок не должен слишком сильно заходить на терминатор, например, достигая его петли. В противном случае он может оказаться достаточно близко к черенку антитерминаторной шпильки, и срываться в результате ее действия, что, конечно же, не согласуется с механизмом КАР, согласно которому полимеразу срывает именно терминатор, а антитерминатор, наоборот, сам полимеразу не срывает, но зато не дает образоваться терминатору.

В заключение приведем несколько реальных примеров 3'-концов последовательностей, начиная с правого плеча терминатора, вместе с использовавшимися наборами параметров полиурацила. В этих примерах нуклеотиды правого плеча терминатора подчеркнуты, выделенные U-богатые участки показаны серым фоном.

GAAGCGGGCUUUUUUGUUUCUAGCUCUUA	по умолчанию
GAAGCGGGCUUUUUUGUUUCUAGCUCUUA	-u3
GGAUGCGGAGGCUUCCCUCUCUCAUC	-lura10 -u0.5
GGAUGCGGAGGCUUCCCUCUCUCAUC	-u3
GGAGGCUUUUUUUGUACCUG	-lura13 -u.69
GGAGGCUUUUUUCGUAUAUGGAUUC	-lura18 -u.61
AUGGGGGGCUUUUUUAUUUGUAGUUAUUUGUAUUAGUAAUCGA	-un2
AUGGGGGGCUUUUUUAUUUGUAGUUAUUUGUAUUAGUAAUCGA	-lura25 -u.7
AUGGGGGGCUUUUUUAUUUGUAGUUAUUUGUAUUAGUAAUCGA	-u3
GAGCGGGUUUUUUUAUUGCCGUUU	-u3 -ug4
AGUCCGGGGGUUUUUUUUACAACUA	-u3 -ug5
GGCCGCCAUCCGCUAGA	-u2 -ug4
GGUCGGUUGCUUUACUUA	-u3 -ug2
CAGGAGGGCCGUACC	-u1 -ug6

8.4 Число траекторий

Для грубой проверки наличия КАР вначале можно запустить программу с параметром `-z100`, однако оценка вероятности терминации при таком числе траекторий будет грубая (достоверным можно считать только первый знак после запятой), и, следовательно, легко прийти к ошибочному общему выводу. Мы рекомендуем делать какие-либо количественные выводы по результатам не менее тысячи траекторий моделирования (параметр `-z1000`).

8.5 Варьирование энергии спиралей

Существующие методы вычисления энергии состояния вторичной структуры используют значения энергии стекинга, определенные весьма грубо (иногда даже с допуском ±50%). Кроме того, есть основания полагать, что не все эффекты взаимодействия в достаточной мере учтены при вычислении энергии, особенно в случае псевдоузлов. В то же время, эксперименты с моделью показали, что во многих случаях даже небольшая (порядка 10%) систематическая ошибка при моделировании баланса равновесия конкурирующих состояний (например, с образованием антитерминатора или терминатора) способна полностью исказить картину зависимости вероятности терминации от концентрации аминокислоты.

Не имея реальных возможностей определить более точные данные или применить более надежные методы вычисления энергии вторичной структуры, мы реализовали в программе возможность внесения вручную поправки (с любым знаком) к энергии одной выбранной спирали, для чего предусмотрены параметры программы `-mn`, `-me`, `-mh` (см. раздел 5). В частности, это помогает скомпенсировать негативные последствия того, что реально существующие структуры не могут образоваться при моделировании, например, ввиду наличия в них гипоспиралей с длиной меньше минимально допустимой. Эксперименты с этой поправкой могут дать дополнительный материал о действительном наличии или отсутствии КАР в конкретной последовательности, равно как и для уточнения способов определения энергии вторичной структуры в тех случаях, когда КАР доказана экспериментально.

8.6 Известные ограничения

В текущей версии программы RNAmodel установлены следующие ограничения технического характера:

- длина указанного имени файла, включая путь: < 128 символов
- длина последовательности: < 300 нт
- число непродолжаемых спиралей в последовательности: < 500
- число пар нуклеотидов в одной шпильке: < 64
- число нуклеотидов в петлях и выпячиваниях одной шпильки: < 256
- число гипоспиралей в одном макросостоянии: <16
- число участков последовательности, где запрещено спаривание: ≤ 6

Список литературы

1. V. Lyubetsky, L. Rubanov, A. Seliverstov, S. Pirogov. Model of gene expression regulation in bacteria via formation of RNA secondary structures. *Molecular Biology*. Vol.40, 3, 2006, p. 440-453.
2. В.А. Любецкий, К.Ю. Горбунов, С.А. Пирогов, Л.И. Рубанов, А.В. Селиверстов. Алгоритм и результаты счета для модели регуляции экспрессии генов у бактерий на основе формирования вторичных структур РНК. *Информационные процессы*. Том 5, 5, 2005, стр. 337-366. http://www.jip.ru/2005/337-366.pdf.
3. В.А. Любецкий, А.В. Селиверстов. Вычисление эффективности регуляции биосинтеза триптофана у бактерий на основе модели классической аттенюации. *Информационные*

процессы. Том 6, 1, 2006, стр. 55-57. http://www.jip.ru/2006/55-57-2006.pdf.

4. Lyubetsky V., Pirogov S., Rubanov L., Seliverstov A. Modeling Classic Attenuation Regulation of Gene Expression in Bacteria. *Journal of Bioinformatics and Computational Biology*. Vol.5, 1, 2007, p. 155-180.
5. L. Rubanov and V. Lyubetsky. RNAmodel Web Server: Modeling Classic Attenuation in Bacteria. *In Silico Biology*. Vol.7, 3, 2007, p. 285-308.
6. A. Xayaphoummine, T. Bucher, H. Isambert, "Kinefold web server for RNA/DNA folding path and structure prediction including pseudoknots and knots", *Nucleic Acids Res*. **33** (Web Server issue), W605–10 (2005).