Биоинформатика

В настоящее время слово биоинформатика стало очень модным, оно употребляется в трех разных смыслах. Первый смысл связывают с телепатией, экстрасенсорикой и т.д., об этом мы говорить не будем. Второй смысл связан с применением компьютеров для изучения любого биологического объекта, но эту тему мы тоже не будем затрагивать. Речь пойдет о биоинформатике в узком смысле слова, а именно о применении компьютерных методов для решения задач молекулярной биологии, в основном анализа разных последовательностей (аминокислотных, нуклеотидных). Эта наука возникла в 1976-1978 годах, окончательно оформилась в 1980 году со специальным выпуском журнала «Nucleic Acid Research» (NAR). Биоинформатика включает в себя:

- базы данных, в которых хранится биологическая информация
- набор инструментов для анализа тех данных, которые лежат в таких базах
- правильное применение компьютерных методов для правильного решения биологических задач

На рисунке показаны соотношение этапов развития биоинформатики (справа) с возникновением разных экспериментальных методик и полученных результатов экспериментальных исследований.

	Технология	Биоинформатика
1962		Молекулярные часы
1965	Секвенирование tRNA	База данных PIR
1970	Обратная транскрипция	Алгоритм выравнивания NW
1972	Клонирование	
1977	Секвенирование	База данных PDB
1980		Спец.выпуск NAR, Базы данных нукл. Послед.
1981		Алгоритм выравнивания SW
1982	Секвенирование ДНК фага лямбда	
1983	PCR	Алгоритм поиска по базе данных WL
1985	Секвенирование ДНК вирусов	FASTA - поиск по базе данных
1987		GeneBank , Профили
1989	Программа "Геном человека"	Swiss-Prot , NCBI
1991	EST	
1992	Первая хромосома дрожжей	BLOSSUM
1993	Автоматическое секвенирование	
1995	Первый геном бактерии	База данных SCOP
1996	Первый геном архейный	
1997		PSI-BLAST, Кластеры ортологичных генов
1998	Геном червя	
2001	Геном человека	

В 1962 году была придумана концепция "молекулярных часов", в 1965 была секвенирована тРНК, определена ее вторичная структура, в это же время были созданы базы данных PIR для хранения информации об аминокислотных последовательностях. В 1972 году было придумано клонирование. В 1978 году были разработаны методы секвенирования, была создана база данных пространственных структур белков. В 1980 был выпущен спецвыпуск журнала NAR, посвященный биоинформатике, затем были придуманы некоторые алгоритмы выравнивания последовательностей, о которых речь пойдет дальше. Дальше был придуман метод ПЦР (полимеразная цепная реакция), а в биоинформатике - алгоритмы поиска похожих фрагментов последовательностей в базах данных. В 1987 году оформился GeneBank (коллекция нуклеотидных последовательностей) и т.д.

Биолог в биоинформатике обычно имеет дело с базами данных и инструментами их анализа. Теперь разберемся, какие базы данных бывают в зависимости от того, что в них помещают. Первый тип – архивные базы данных, это большая свалка, куда любой может поместить все, что захочет. К таким базам относятся

- GeneBank & EMBL – здесь хранятся первичные последовательности
- PDB – пространственные структуры белков,

и многое другое.

В качестве курьеза могу привести пример: в архивной базе данных указано, что в геноме археи (архебактерии) есть ген, кодирующий белок главного комплекса гистосовместимости, что является полной чепухой.

Второй тип – курируемые базы данных, за достоверность которых отвечает хозяева базы данных. Туда информацию никто не присылает, ее из архивных баз данных отбирают эксперты, проверяя достоверность информации – что записано в этих последовательностях, какие есть экпериментальные основания для того, чтобы считать, что эти последовательности выполняют ту или иную функцию.

К базам данных такого типа относятся:

- **Swiss- Prot** – наиболее качественная база данных, содержащая аминокислотные последовательности белков
- **KEGG** – информация о метаболизме (такая, которая представлена на карте метаболических путей, которую те, кто ходит на лекции, видели на лекции № 2)
- **FlyBase** – информация о *Drosophila*
- **COG** – информация об ортологичных генах.

Поддержание базы требует работы кураторов или аннотаторов. Тем не менее, даже в курируемых базах данных могут встречаться курьезные надписи, например такая забавная надпись:

По крайне мере здесь кураторы базы данных честно признаются, что не знают, как это случилось.

Третий тип – производные базы данных. Такие базы получаются в результате обработки данных из архивных и курируемых баз данных. Сюда входит:

- SCOP – База данных структурной классификации белков (описывается структура белков)
- PFAM – База данных по семействам белков
- GO (Gene Ontology) – Классификация генов (попытка создания набора терминов, упорядочивания терминологии, чтобы один ген не назывался по разному, и чтобы разным генам не давали одинаковые названия)
- ProDom – белковые домены
- AsMamDB – альтернативный сплайсинг у млекопитающих

И интегрированные базы данных, в которых вся информация (курируемая, не курируемая) свалена в кучу, и введя имя гена, можно найти всю связанную с ним информацию – в каких организмах встречается, в каком месте генома локализован, какие функции выполняет и т.д.

- **NCBI Entrez** – доступ к информации о нуклеотидных и аминокислотных последовательностях и структурах
- **Ecocyc** – все о *E. coli* – гены, белки, метаболизм и пр.

Теперь перейдем к рассмотрению инструментов биоинформатике. Инструменты определяются задачами, которые мы хотим решать.

Основу биоинформатики составляют сравнения. Если у нас есть, например, аминокислотная последовательность, о которой у нас есть экспериментальные данные, и известны ее функции, и другая, похожая на нее последовательность, мы можем предположить, что эти последовательности выполняют сходные функции. Это задача поиска сходства последовательностей

Другая задача связана с анализом генома. Недавно было объявлено, что полностью просеквенирован геном человека, но так же просеквенировали геномы и других организмов: три генома растений, мыши, крысы, кошки, собаки, курицы, рыбы, лягушки завершается, шимпанзе завершается, две дрозофилы сделаны, малярийный комар, червяки, дрожжи и т.д. – всего около 30 видов эукариотических геномов. Также просеквенированы сотни бактериальных геномов. Один бактериальный геном можно просеквенировать в хорошо оборудованной лаборатории за неделю. При этом получают длинную нуклеотидную последовательность нуклеотидов. Там есть гены – белок-кодирующие участки, и участки, кодирующие тРНК и рРНК. Возникает задача найти эти гены. Другая задача – поиск сигналов в ДНК, то есть тех участков ДНК, которые отвечают за регуляцию - сайты связывания регуляторных белков, элементы вторичной структуры мРНК, которая транскрибируется с этого гена и др.

Есть задача предсказания вторичной структуры РНК. А также есть большой класс задач анализа белков. Для решения этих задач надо создавать методы анализа, то есть алгоритмов (протоколов) и программ для анализа. При создании метода надо иметь критерий того, что метод адекватен, соответствует реальности.

Как оценить "правильность" метода? Геном типичной бактерии содержит около 1000 генов. Как уже упоминалось, секвенировать геном можно за неделю. Экспериментальная характеристика одного белка требует как минимум 2 месяца работы современной лаборатории.

Для того, чтобы определить, насколько предложенный метод анализа хорош и правилен, существует так называемый «золотой стандарт». Например, у нас есть метод определения генов. Если после его применения на какой-либо последовательности, в которой известно месторасположение генов, наши результаты совпадают с тем, что есть на самом деле на 80-90%, значит наш метод правильный и эффективный. В этом и заключается суть «золотого стандарта».

Или предсказание вторичной структуры РНК. Экспериментально ее определить очень трудно, но есть РНК, структура которых хорошо известна – это рРНК и тРНК. И если наш метод хорошо предсказывает структуру этих известных РНК, то можно ожидать, что и для других РНК он будет давать хорошие предсказания.

Вернемся к первой задаче – сравнению последовательностей. Запишем одну последовательность под другой.

attgt**AC**c**TC**g**T**g**G**-**AA**----

-----**AC**-**TC**a**T**a**G**c**AA**ccag

Нам надо при сравнении найти наилучший вариант, так выровнять эту пару последовательностей, чтобы количество совпадений будет максимальным (парное выравнивание). Качество выравнивания оценивают, назначая штрафы за несовпадение букв и за наличие пробелов (когда приходится раздвигать одну последовательность для того, чтобы получить наибольшее число совпадающих позиций).

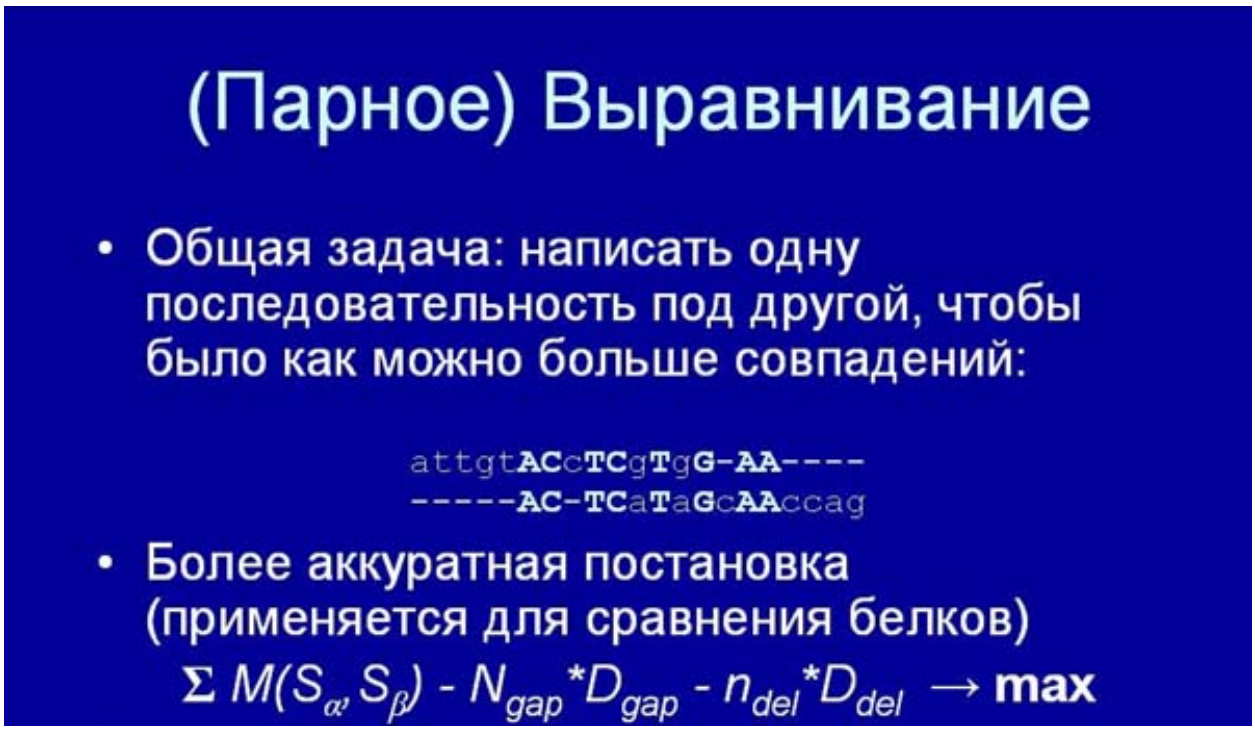
Таким образом, первым делом после секвенирования последовательности ищут в базах данных похожие последовательности, чтобы после сравнения судить о том, какие функции несет эта последовательность. Если две буквы совпали, значит они находятся под давлением отбора, они функционально важны. Известно, что аминокислоты различаются по своим свойствам, поэтому если произошла аминокислотная замена, это может почти никак не повлиять на работу белка, а может сильно его изменить.

Например, если лизин (положительно заряженная аминокислота заменится на лейцин (похожий по созвучию, но совершенно несходный по свойствам), то для пространственной структуры и функций белка это может оказаться катастрофой. А вот замена лизина на аргинин (также положительно заряженный) может не сказаться на структуре белка.

Поэтому при сравнении аминокислотных последовательностей учитывают также матрицу сопоставления аминокислотных остатков (похожих, менее похожих и совсем непохожих).

Как осуществляется выравнивание? Пишем одну последовательность под другой.

Сколько есть способов написать одну последовательность S1 длиной m под другой – S2 длиной n (со вставками)? Об этом можно доказать теорему – попробуйте.



Построим выборочную последовательность S длиной *m+ n* следующим образом: возьмем несколько символов из последовательности S1, потом несколько символов из последовательности S2 потом опять несколько символов из S1, потом опять несколько из S2.

– Каждой выборочной последовательности S соответствует выравнивание и по каждому выравниванию можно построить выборочную последовательность. ***(Доказать!)***

– Количество выборочных последовательностей равно

$$N_{sel} = C_{n+m}^{m} = \frac{(n+m)!}{(n!)(m!)}$$

(Доказать!)

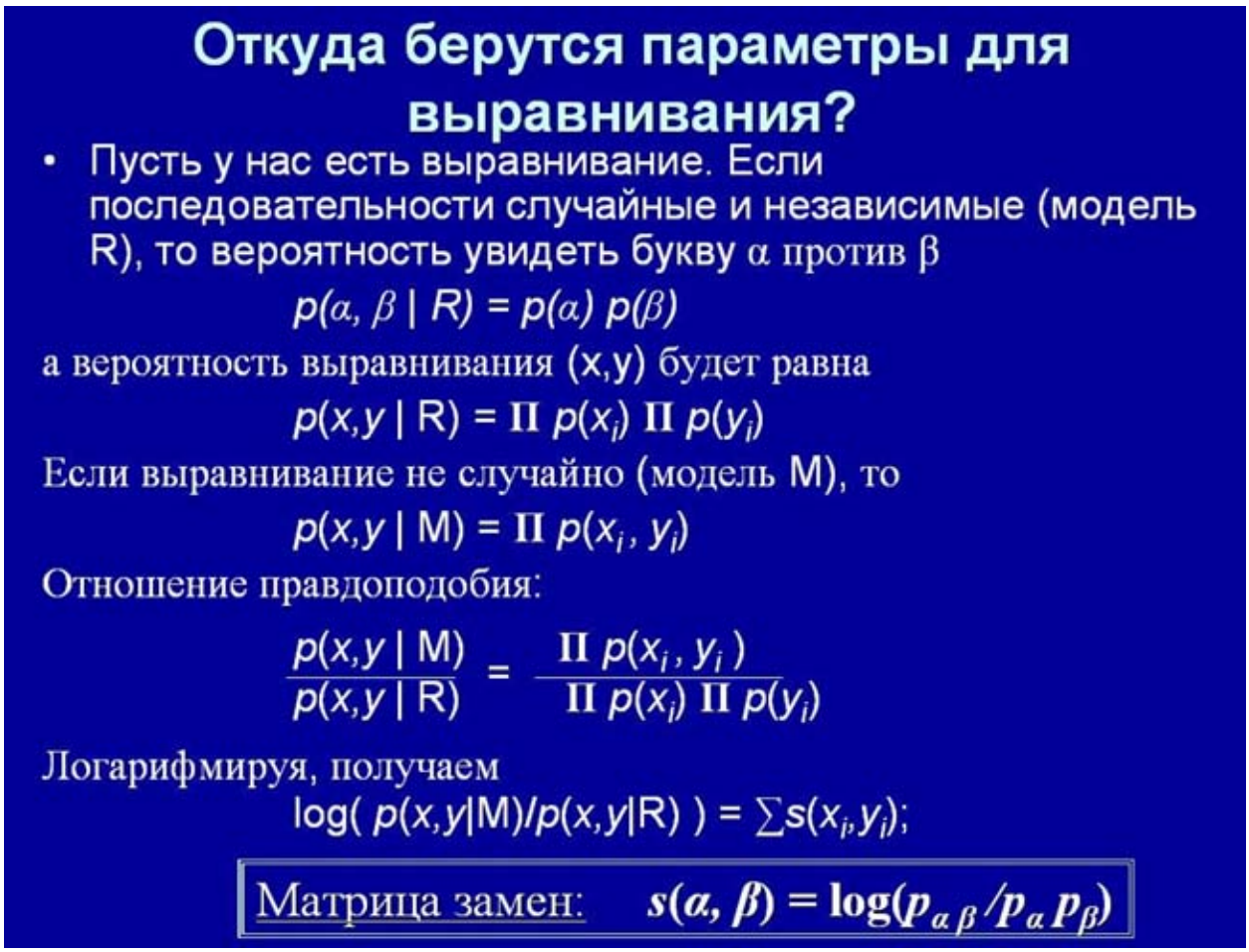
Таким образом количество выравниваний можно определить по формуле:

$$N_{algn} = C_{2n}^{n} = \frac{(2n)!}{(n!)^2} \approx \frac{2^{2n}}{\sqrt{\pi n}}$$

А как же найти оптимальное среди такого большого количества? Можно, конечно, попробовать разные способы, но оказывается, что этот поиск сводится к задаче поиска оптимального пути на графе. Задача поиска оптимального пути на графе решается методами динамического программирования следующим образом. Мы пишем одну последовательность над другой. И у нас есть некая ячейка, в которой мы будем хранить вес наилучшего выравнивания префиксов (то фрагментов

последовательности от начала до данного места). И если у нас известен вес наилучшего выравнивания в 3 ячейках (см. слайд ниже), то мы можем определить вес наилучшего выравнивания в четвертой ячейке. То есть, для того, чтобы найти вес оптимального выравнивания, нам надо просмотреть m*n ячеек (количество ячеек в прямоугольной матрице MxN). Как принято говорить в информатике, это – квадратичный алгоритм. Он занимает время и объем памяти, пропорциональный квадрату длины последовательности. И вместо случайного перебора большого числа вариантов, мы решаем задачу довольно быстро.

Откуда берутся матрицы замен? Мы берем некоторое количество выравниваний, в которое по тем или иным причинам верим, и смотрим, как часто у нас происходят такие замены. Тогда матрица замен является логарифмом отношения некоторых вероятностей, которые можно оценить как частоты.



Итак, у нас имеется замечательный квадратичный алгоритм поиска сходства. Время решения задачи выравнивания пропорционально L1*L2. Мы сравниваем имеющуюся у нас последовательность с последовательностями в банке. L1 = размер банка = 10^8, а для генома человека $3x10^9$. Сравниваемая последовательность обычно имеет размер L2=10^3, количество операций примерно равно 100*1011=1013.) Обычный компьютер имеет быстродействие около 109 операций в сек. На каждый шаг надо ≈102 операций. Тогда время работы равно T≈106 сек ≈ 11 дней. То есть, просеквенировав бактериальный геном из 3000 генов (приблизительно за неделю), на то, чтобы его охарактеризовать, мы потратим 11*3000 дней, то есть проанализировать дольше чем секвенировать, что, конечно, не очень хорошо.

Решением является то, что мы до применения методов динамического программирования сначала выбираем правильных кандидатов для сравнения. Есть такая программа BLAST (basic local alignment search tool), которую все биологи очень любят, она почти правильная. То есть она почти всегда работает так, как требует "золотой стандарт".

Основная идея ее работы заключается в хешировании. В самом начале мы один раз проходим по всему банку и для каждого короткого слова с заранее зафиксированной длиной мы запишем список позиций, где оно встречается в банках.

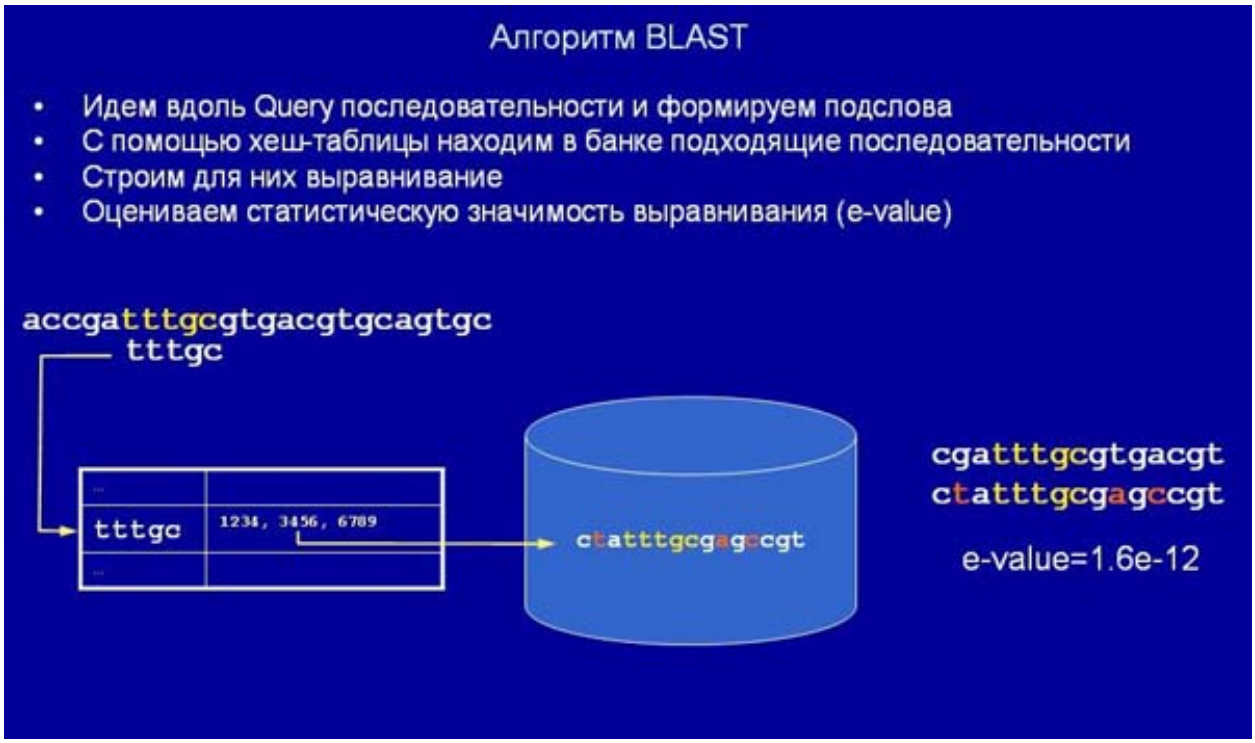
Здесь показано для слов длиной 4, в реальности слова берут не длиной 4, как показано на рис., а длиной 7 или 10 или 13, но принцип тот же. В каких-то случаях "слову" соответствует три позициями, в других – 100 позиций.

Дальше мы идем вдоль последовательности "Query" (та последовательность, которую мы хотим прогнать по банку) и выбираем очередные слова. Смотрим в таблице, где встречается это слово, вытягиваем найденные последовательности из банка и строим выравнивание их с нашей исходной последовательности. Это делается быстро, так как мы сравниваем нашу последовательность не со всеми последовательностями из банка, а только те, которые соответствуют нашему "слову" (tttgc в показанном случае). И выравнивание строим тоже не так аккуратно, как это делает алгоритм динамического программирования, а используем упрощенную схему.

Затем мы оцениваем статистическую значимость этого выравнивания – так называемую e-value. Вообще, есть два понятия, которые очень часто встречаются в биоинформатике: e-value и p-value. E-value – это сколько мы ожидаем увидеть совпадений с таким весом (то есть такого качества), если бы у нас наши последовательность и банк были случайными. Если они случайные, то мы ожидали бы увидеть e^{-1} ^{2} совпадений.

e-value – это ожидаемое число событий, может быть больше единицы. Если e-value маленькое, то, значит, совпадение значимое, и оно несет большую

биологическую информацию. P-value – это вероятность встречи такого соответствия (не может быть больше единицы). При оценке e-value, да и вообще при любых статистических оценках, важно, какая модель лежит в основе всего этого дела. Модель, которая лежит в основе e-value, конечно же, неправильная, потому что мы не знаем правильность статистических характеристик биологических последовательностей. E-value просто дает нам ориентир, и реально, если мы имеем e-value порядка 10^{-2}, то это, как правило, мусор, незначимое соответствие. Правда, есть некоторые специалисты с такой интуицией о структуре белков, которые могут работать с выравниваниями с e-value даже порядка 1. А обычно если исследователи видят e-value > 10^{-3}, они с этим не работают.



Есть разные модификации BLAST: BLASTp (выравнивание аминокислотных последовательностей), BLASTn (выравнивание нуклеотидных последовательностей), BLASTx (выравнивание всех возможных транслятов нашей нуклеотидной последовательности против банка аминокислотных последовательностей), TBLASTx (выравнивание всех возможных транслятов нашей нуклеотидной последовательности против всех транслятов банка нуклеотидных последовательностей). Еще нужно знать, что Nr Data Base – (non redundant) - это база, против которой обычно прогоняют BLAST, в которой нет повторяющихся последовательностей, из которой убраны дубли для того, чтобы не гонять BLAST по одним и тем же последовательностям. И score – это вес выравнивания.

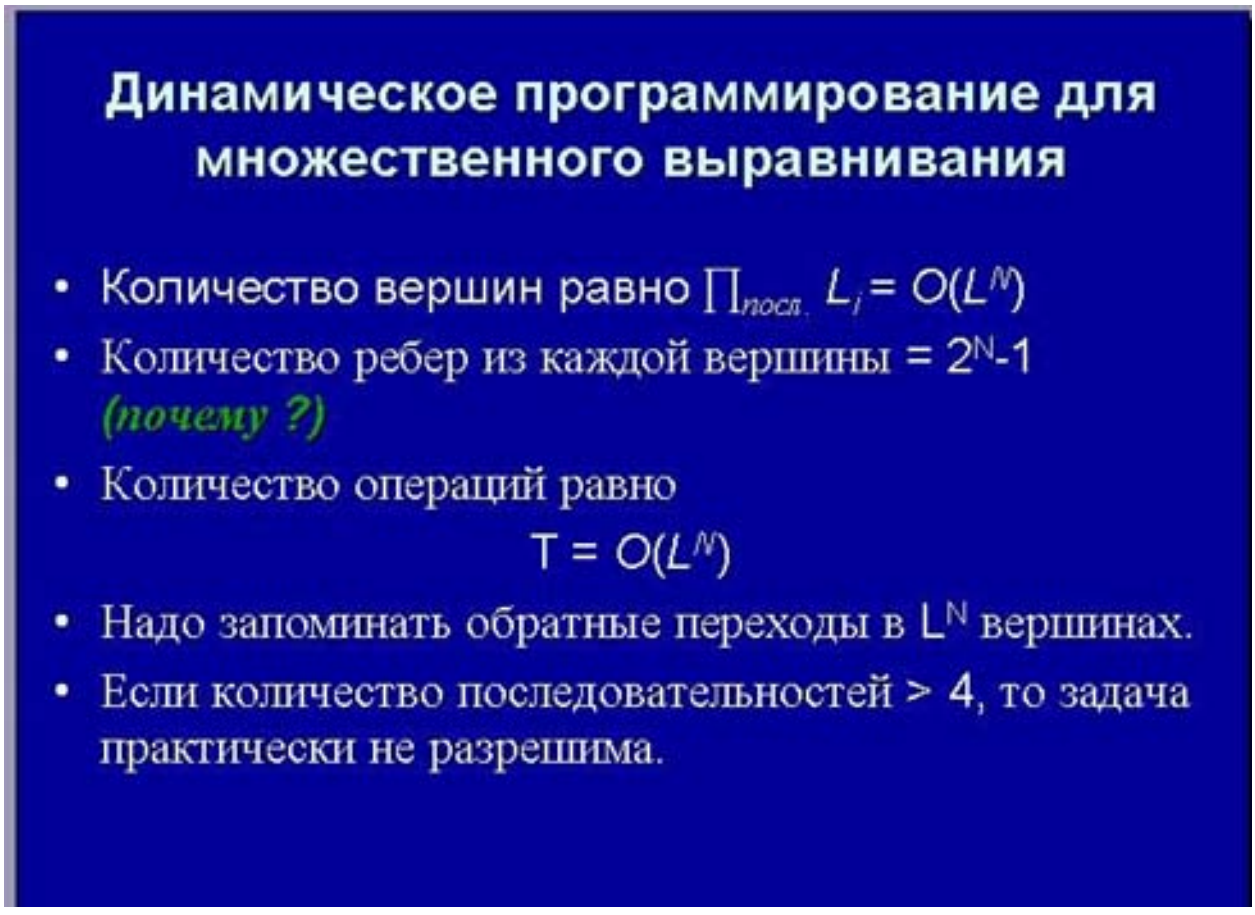
А если на нашу последовательность при поиске налипло, например, не одна, а двадцать последовательностей. При этом возникает задача написать все эти последовательности друг под другом, чтобы увидеть, в какой мере они совпадают, что консервативно (устойчиво повторяется), а что нет, и как устроена наша аминокислотная последовательность. Эта задача называется

Множественное выравнивание

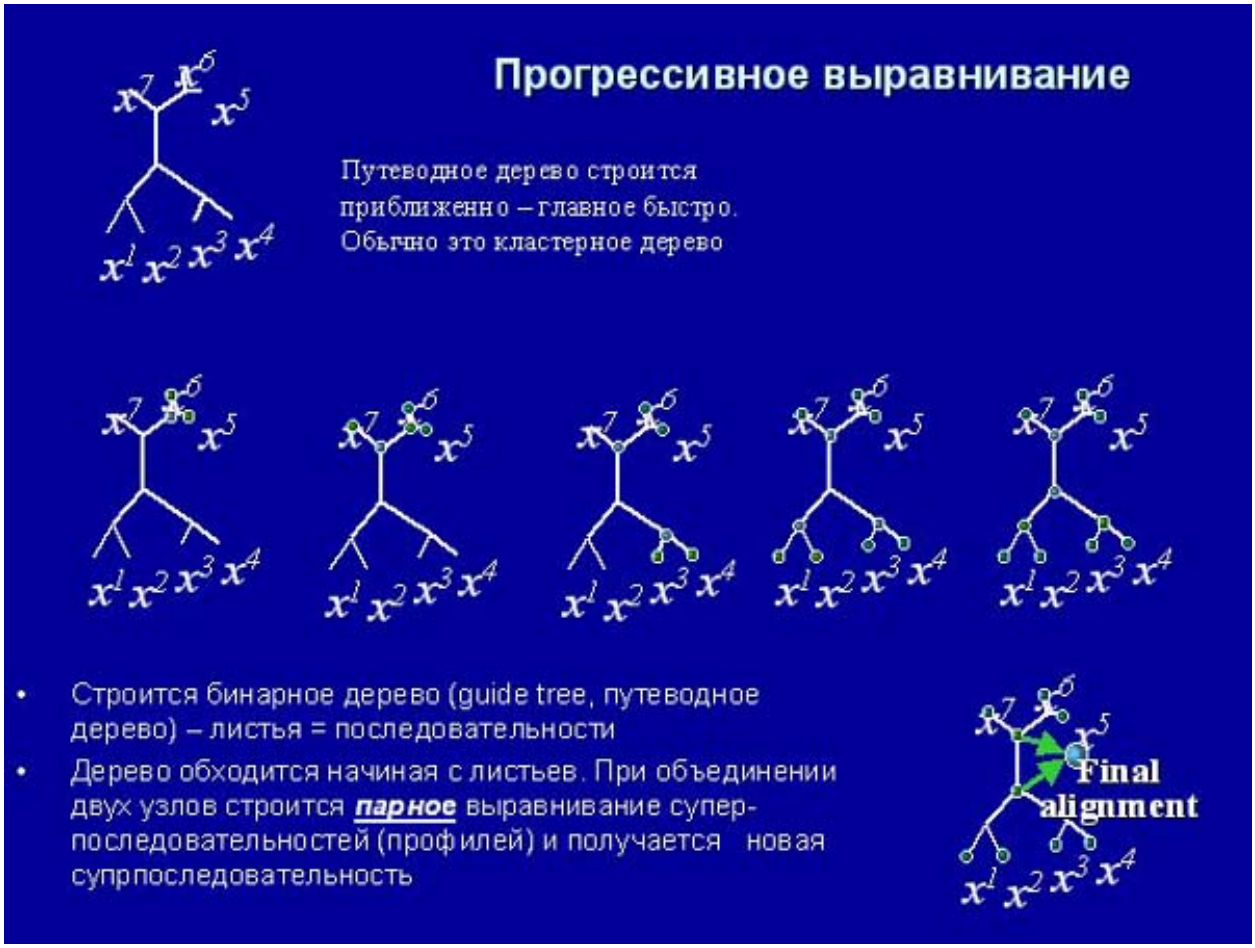
Множественное выравнивание – это такой способ написания нескольких последовательностей друг под другом (может быть, с пропусками в каких-то позициях в разных последовательностях) , чтобы в каждом столбце стояли гомологичные позиции.

Для этой задачи тоже есть «золотой стандарт». Это выравнивание, которое бы получилось, если бы мы выровняли друг под другом последовательности, которые имеют одинаковую пространственную структуру. То есть две экспериментально установленные пространственные структуры белка сопоставляем и отмечаем, какие аминокислотные остатки друг под другом встали (эти остатки соответствуют гомологичным позициям). Это – биологически обоснованное выравнивание. Возникает задача - найти способ (построить алгоритм и определить параметры), который выравнивает последовательности "золотого стандарта" (то есть последовательности, для которых пространственная структура известно) правильно. Если такой алгоритм построен, то есть надежда, что он выровняет последовательности с неизвестной пространственной структурой тоже правильно.

Для решения задачи множественного выравнивания можно попробовать написать многомерную матрицу и построить методом динамического программирования с просмотром многомерной матрицы. Тогда количество вершин будет порядка L^n , где L – длина, а n – количество последовательностей. Так как типичное количество последовательностей в семействе белков сотни, то 300 аминокислот дадут 300^{100} – это очень много, этот алгоритм для множественного выравнивания не подходит.



Тогда придумали метод прогрессивного выравнивания. Зная расстояния между любой парой последовательностей, мы можем построить выравнивание, определить вес выравнивания, и построить какое-то бинарное дерево. Затем мы обходим это дерево, последовательно проводя парные выравнивания наиболее близких последовательностей. Объединяем, получаем выравнивание. Соединяем суперпоследовательности, получаем следующее выравнивание. В конце концов получаем выравнивание в корне.



Такое постепенное построение выравнивание решает задачу, которую мы не можем сформулировать математически. В биоинформатике очень часто нельзя построить математическую формулировку задачи, которую мы решаем. Поэтому формулировка задачи, которую решает алгоритм BLAST, выглядит так: мы находим то, что находит программа BLAST. Также мы не можем сказать, что мы оптимизируем при множественном выравнивании.

Одна и та же биологическая задача может приводить к разным математическим постановкам одной и той же задачи. Есть примеры, когда одна и та же задача может быть построена так, что она будет математически решаемой или математически не решаемой. Есть класс задач, для которых не существует хороших алгоритмов. Но при построении множественных выравниваний мы решаем с помощью данного алгоритма, без формулировки математической задачи.

Дальше идет задача

предсказания вторичной структуры РНК

Вторичная структура РНК – структура, образуемая спаренными основаниями на однонитевой молекуле РНК. Биологическая роль вторичной структуры: структурная (РНК – рибосомная, тРНК), регуляция (рибопереключатели, аттенюация, микроРНК), рибозимы, стабильность РНК.

На рисунке показана типичная вторичная структура РНК и разные формы представления вторичной структуры РНК:

Вся РНК состоит из петель и спиралей (указано на рисунке). Петли бывают следующих типов: шпилька, внутренняя, выпячивание, множественная, псевдоузел. Так вот, возникает задача установить, кто с кем спарен. Биологическая формулировка этой задачи звучит так: дана последовательность РНК, определить ее правильную вторичную структуру. «Золотой стандарт» - тРНК и рРНК. Количество возможных вторичных структур очень велико. Задачу можно сформулировать таким образом (законным с точки зрения физики): надо минимизировать энергию, поскольку правильная вторичная структура наиболее стабильная. На самом деле, с точки зрения биологии это не совсем верно, но формулировка очень удобная с точки зрения физики и математики. Далее вопрос, что оптимизировать и как оптимизировать.

Предположим, что мы не будем минимизировать усилия по поиску, а все переберем. Построим такой граф, в котором вершины – потенциальные спирали, а ребра проводятся, если две потенциальные спирали в вершинах совместимы (то есть, если две спирали могут одновременно существовать в данной молекуле РНК).

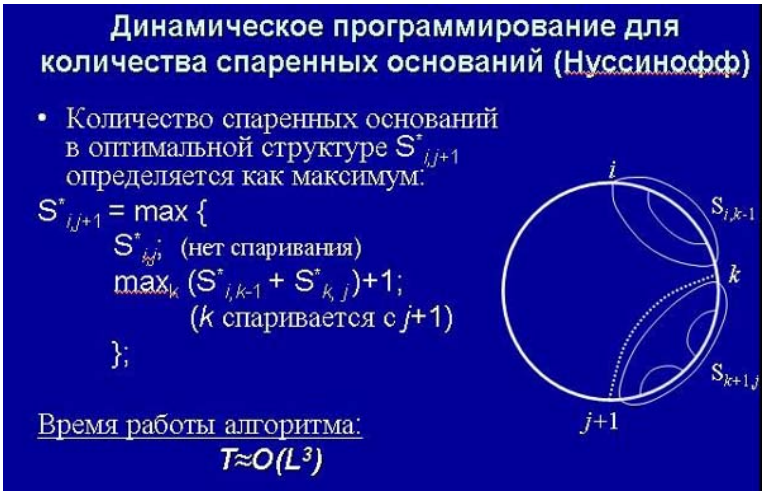
Тогда вторичной структурой будет любой полный подграф, то есть такой граф, в котором все вершины между собой соединены – называется "клика". Тогда задача такова: в таком графе найти клику. Клика будет соответствовать хорошей структуре.

Но, к сожалению, задача поиска клики в графе является математически плохой – для нее, скорее всего, не существует эффективного алгоритма ее решения (кроме полного перебора всех вариантов).

Если мы fgh уберем, то получим клику, некую вторичную структуру. Можем получить и другую клику.

Вторичная структура может быть представлена в виде правильной скобочной структуры, как на рисунке ниже. Левая часть – открывающая скобка, правая часть – закрывающая скобка. Вторичная структура тоже может быть представлена в виде дерева, но важно, что количество возможных структур порядка $1{,}8^L$ (это доказывается в теореме, которую я не буду здесь представлять). Это тоже очень много, поэтому задача поиска клики тоже не эффективна.

Тем не менее, есть алгоритм динамического программирования, который позволяет нам найти за кубичное (а не квадратичное, как раньше) время найти структуру, имеющую наибольшее количество спаренных оснований. Основная идея его (как и любого алгоритма динамического программирования) заключается в том, что если мы знаем все решения на какой-то части, то мы можем сказать, какое будет решение на чуть большем фрагменте.



Можно минимизировать не число спаренных оснований, а минимизировать энергию (эта задача сложнее, но ее с помощью разных ухищрений тоже можно оставить кубичной). Минимизация все равно не позволяет достигнуть большой точности предсказания. Проблемы предсказания вторичной структуры РНК.

Только около 65-70% тРНК сворачиваются в правильную структуру.

Для предсказания вторичной структуры используются энергетические параметры, а они определены не очень точно. Более того, в клетке бывают разные условия, и, соответственно, реализуются разные параметры.

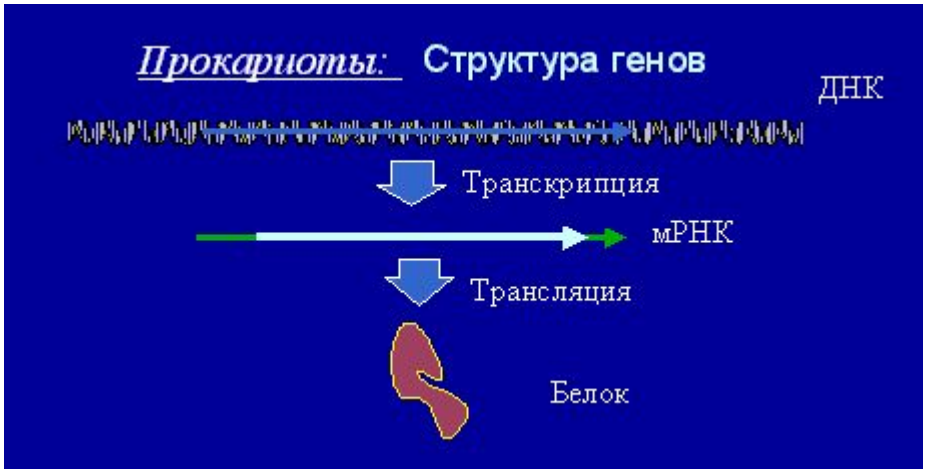
Находится единственная структура с минимальной энергией, в то время как обычно существует несколько структур с энергией, близкой к оптимальной.

Поэтому есть предложения искать субоптимальные структуры и искать эволюционно консервативные структуры (структуры тРНК и рРНК определены именно так). То есть забыть про энергию, и если мы знаем, что эти наборы РНК выполняют одну и ту же функцию, то мы можем построить такую структуру, которая была бы общей для всех этих последовательностей.

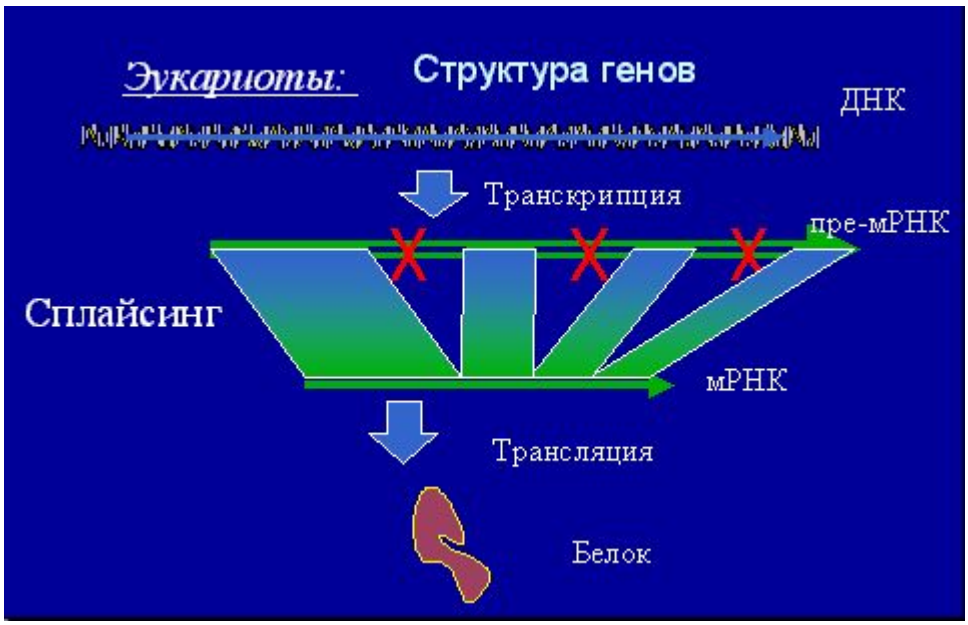
Теперь я расскажу, как это все применяется.

Исследование консервативности альтернативного сплайсинга, или Почему мышь не стала человеком?

Структура генов прокариот очень проста: есть начало, есть конец, получается мРНК, которая имеет начало и конец, идет транскрипция, трансляция и белок.



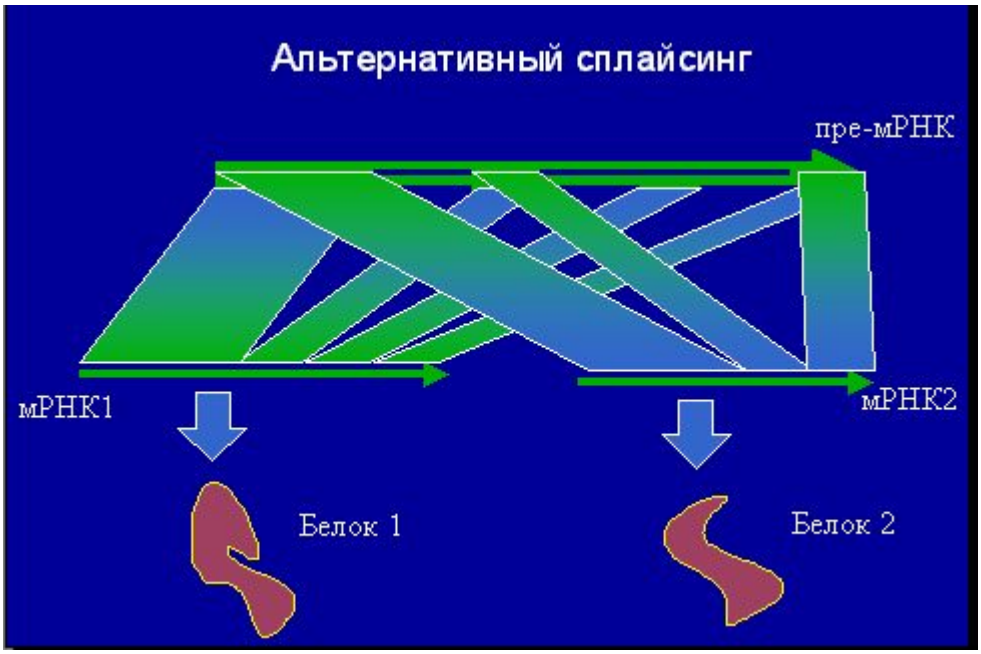
У эукариот структура гена сложнее. Из длинной мРНК удаляются (вырезаются) **интроны** (insertion sequences, вставочные последовательности), а оставшиеся экзоны сшиваются в единую нить. Из пре-мРНК получается зрелая мРНК, процесс называется **сплайсингом**. Потом происходит трансляция зрелой мРНК, в результате образуется белок. Мы будем интересоваться экзонами и интронами.



Если бы мы умели правильно предсказывать интроны и экзоны, мы бы могли разметить ген на белок-кодирующие и белок-некодирующие участки.

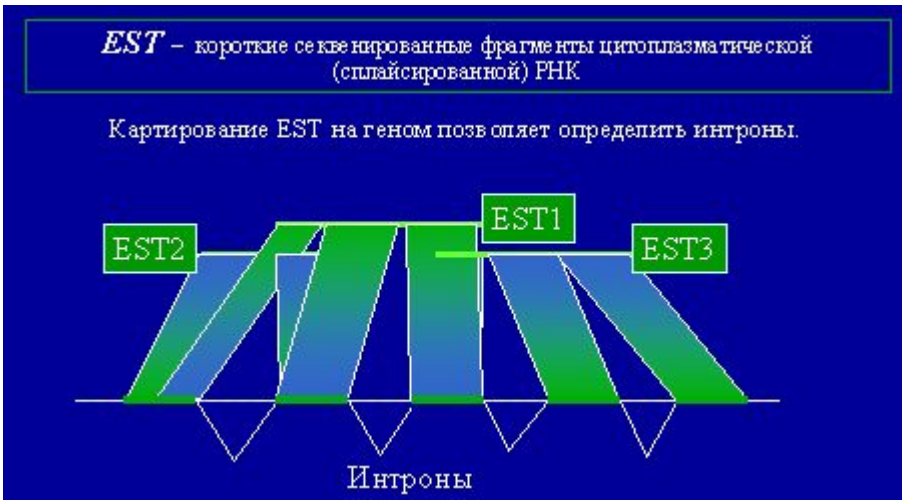
Альтернативный сплайсинг

Оказывается, ситуация еще сложнее. РНК, прочитанная с одного и того же гена, может сплайсироваться по-разному, что приводит к образованию мРНК с разными наборами экзонов: какой-то экзон в один вариант мРНК попадает, а в другой - нет, и в итоге получатся две разных мРНК и, соответственно, два разных белка. Это называется альтернативным сплайсингом. Таким образом, на уровне созревания мРНК могут образовываться разные РНК-продукты, которые приводят к образованию разных белков.



Сплайсинг происходит в ядре, трансляция – в цитоплазме. Для изучения того, что же оказалось в цитоплазме (то есть того, что подвергается трансляции), секвенируют короткие, 500-600 до 1000 нуклеотидов куски цитоплазматической

РНК. Такие сиквенсы называются EST (expresstion sequence tag – "ярлыки экспрессируемых последовательностей"). EST – это короткие, прочитанные однократно (то есть весьма неточно), фрагменты цитоплазматической (сплайсированной, содержащей только экзоны) РНК. Если у нас есть геном, то мы можем эти EST картировать на геном и, тем самым, найти, где находятся интроны и экзоны.



Если при картирование EST полностью, без перерывов, соответствует геномной последовательности – это ген без интронов. Если EST ложится на геном с перерывами, то мы наблюдаем результат сплайсинга. Если же разные EST демонстрируют несколько вариантов расположения в одном и том же участке генома (то есть выявляют разные сочетания экзонов), то мы наблюдаем альтернативный сплайсинг. Экзон, который может включаться в белок, а может и не включаться, называется кассетным экзоном. мРНК с разными наборами экзонов данного гена (то есть в которые некий кассетный экзон или включается или не включается), называются изоформами.

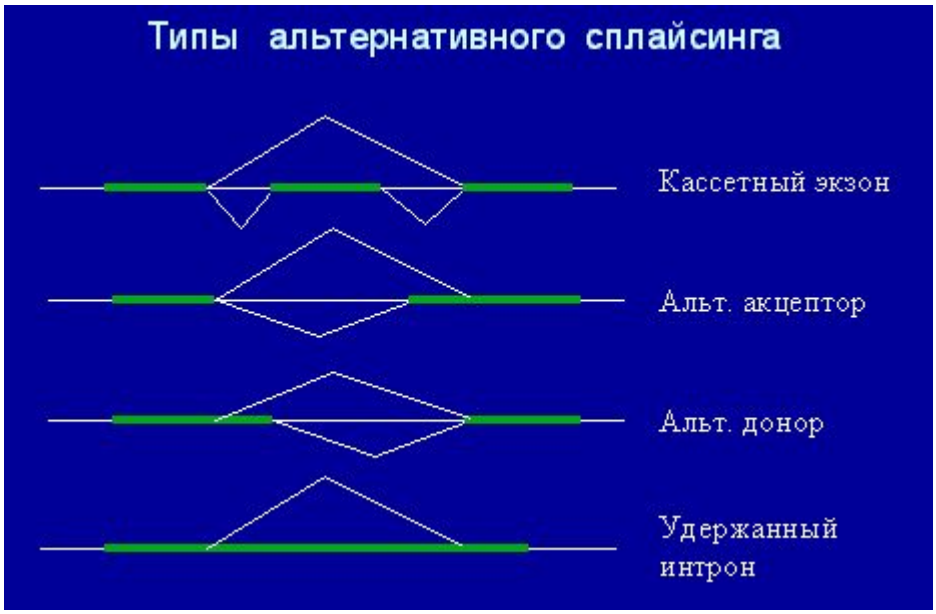
Частота альтернативного сплайсинга

Сначала альтернативный сплайсинг был обнаружен у вирусов, считалось, что это экзотика. До 1998 г. считалось, что только около 6% генов человека имеют альтернативный сплайсинг. Рассчитали, что для того, чтобы обеспечить наблюдаемое разнообразие белков, в геноме человека должно было быть 80 – 100 тысяч генов. В 1998 году было показано, что около половины генов человека имеют альтернативный сплайсинг. За счет альтернативного сплайсинга число генов может быть меньше числа кодируемых ими белков, так как с одного гена может образовываться несколько белков.

Как было написано в какой-то газете "Многолетними усилиями ученых количество генов человека было сокращено со 100 тысяч до 25". Действительно, по последним оценкам в геноме человека около 25-30 тысяч генов. Оценка количества белков не изменилась - разных белков около 80-100 тысяч. Разнообразие белков обеспечивается альтернативным сплайсингом. Например, в одних клетках белок должен быть в цитоплазме, в других - такой же белок в мембране, в третьих – транспортироваться наружу. И это легко делается не за счет наличия разных генов для каждого случая, а за счет альтернативного сплайсинга , который цепляет на N-конец разные сигналы, при том что "рабочая часть" белка остается одной и той же, и одна изоформа белка размещается в мембране, другая изоформа белка – в цитоплазме, и т.д.

Сейчас общеизвестно, что не менее 50% генов человека альтернативно сплайсируется.

Альтернативный сплайсинг бывает разных типов (галочками показано, как вырезаются экзоны):



На этом рисунке показаны кассетный экзон (вставляемый в одни изоформы и отсутствующий в других), альтернативный акцептор, альтернативный донор, далее интрон может либо вырезаться, либо не вырезаться.

Теперь вернемся к вопросу о человеке и мыши. Человек и мышь биологически очень похожи. Белки похожи – уровень сходства аминокислотных последовательностей 80%, также похожа значительная часть некодирующих областей генома. Практически у всех генов одинаково устроена экзон-интронная структура, для 99% генов экзонная структура одинакова. Только 1% генов уникален у каждого генома, остальные гены имеют гомологи в другом геноме. Интересен тот факт, что при таком относительно невысоком уровне различий человека от мыши внешне отличают легко. А два вида мухи дрозофилы вряд ли кто-то различит на глаз, хотя генетически они различаются сильнее, чем человек и мышь.

Возникает вопрос: Если геномы одинаковы, то может быть, и белки одинаковы? Непонятно, чем же человек отличается от мыши. Одинаково ли устроен альтернативный сплайсинг у мыши и человека?

Наивный подход для ответа на этот вопрос такой: возьмем весь набор альтернативного сплайсинга мыши и человека и сравним его. Этот подход неправильный, так как при исследовании альтернативного сплайсинга мы здесь имеем дело с EST. Если у человека EST просеквенировано несколько миллионов штук, то у мыши сделано всего несколько тысяч, поэтому там, где мы можем увидеть альтернативный сплайсинг у человека, можем ничего не увидеть у мыши, так как базы данных еще не совсем полные. Поэтому такое сравнение даст нам неправильный ответ.

Правильный подход в данной ситуации заключается в следующем: мы на основе имеющихся данных на человеческой ДНК строим мРНК, соответствующую белку, и затем этот белок проецируем на мышиный геном. Если оказывается, что для этого белка (или его части) нет кодирующих последовательностей в мышиной ДНК, то это значит, что тот экзон, который есть у человека, отсутствует в геноме у мыши.

Геном	человек	мышь
Гены с альт.сплайсингом	126	124
Кассетные экзоны	177	123
Альтернативный акцептор	51	46
Альтернативный донор	52	53
Сохраненный интрон	12	29
Всего альтернатив	285	252

Возьмем человеческие и мышиные гены, происходящие от общего предкового гена Возьмем такие пары генов-ортологов, сделаем сравнение. Мы получим некоторую выборку, среди которым 50% генов человека имеют такие изоформы, которых нет у мыши, то же самое и с мышью.

Сравним пары генов человек-мышь. Например, ген бета-глобина человека и мыши – такие гены, разошедшиеся в процессе эволюционного видообразования, называются ортологами. Выборку мы взяли не очень большую, в ней присутствовали гены, имеющие альтернативный спалйсинг. И оказалось, что у 52% человеческих генов есть такие экзоны, которых нет у мыши. И половина мышиных генов имеет такие изоформы, которых нет у человека.

Результаты

Геном	человек		мышь	
Консервативность	C	NC	C	NC
Кассетные экзоны	74	26	39	9
Альт. донор	16	10	17	6
Альт. акцептор	19	15	16	9
Сохр. интрон	5	0	10	4
Всего альтернатив	114	51	82	28
	69%	31%	75%	25%
генов	41	44	30	26
	48%	52%	54%	46%

Но нам могут сказать – вы использовали EST, это неточные данные. Если мы возьмем полноразмерные мРНК (а данные по ним гораздо точнее, хотя общее количество сиквенсов по ним меньше), и проведем с ними ту же процедуру, то окажется, что примерно треть генов человека имеет изоформы, которые в геноме мыши не кодируются, отсутствуют, и также в геноме человека отсутствуют мышиные экзоны.

Контроль

- Отобраны альтернативы, подтвержденные секвенированием мРНК

Геном	человек				мышь			
Консервативность	C		NC		C		NC	
Кассетные экзоны	74	56	26	25	70	39	9	5
Альт. донор	16	18	10	7	24	17	6	6
Альт. акцептор	19	13	15	5	15	16	9	6
Сохр. интрон	5	4	0	3	8	10	4	7
Всего альтернатив	114	96	51	30	117	82	28	24
	69%	76%	31%	24%	83%	75%	25%	17%
генов	41	45	44	28	68	30	26	22
	48%	62%	52%	38%	76%	54%	46%	24%

А вот конкретные примеры: сверху изображены ДНК и ее изоформы у человека, а снизу – то же у мыши. Например, для белка p53, который участвует в регуляции клеточных процессов (раковое перерождение, апоптоз). Видно, что у мыши есть изоформа, которая теряет экзон, порождая стоп в другом месте.

Представленные данные показывают, что альтернативный сплайсинг – явление весьма распространенное, и что мышь сильно отличается от человека по альтернативному сплайсингу. Можно сделать и эволюционное предположение. По-видимому, альтернативный сплайсинг допускает большую свободу для создания новых белков, или изменения функций существующих белков, и в этом и состоит его связь с эволюцией.