МОЛЕКУЛЯРНАЯ БИОЛОГИЯ, 2007, том 41, № 5, с. 918–925

МАТЕМАТИЧЕСКАЯ И СИСТЕМНАЯ БИОЛОГИЯ

УДК 575.852

РЕКОНСТРУКЦИЯ ПРЕДКОВЫХ РЕГУЛЯТОРНЫХ СИГНАЛОВ ВДОЛЬ ДЕРЕВА ЭВОЛЮЦИИ ФАКТОРА ТРАНСКРИПЦИИ

© 2007 г. К. Ю. Горбунов*, В. А. Любецкий

Институт проблем передачи информации им. А.А. Харкевича Российской академии наук, Москва, 127994

Поступила в редакцию 19.10.2006 г.
Принята к печати 28.01.2007 г.

Предлагаются модель и алгоритм, которые позволяют реконструировать предковые регуляторные сигналы, прежде всего, взаимодействия белок-ДНК, во внутренних вершинах дерева эволюции фактора транскрипции, опираясь на современное распределение этих сигналов. Одновременно алгоритм предлагает эволюционный сценарий – набор ребер в этом дереве, на которых сигнал в ходе эволюции изменился наиболее значительно. Модель и алгоритм тестировали, привлекая искусственные данные и также биологические данные (сигналы NrdR и MntR, LacI-семейства).

Ключевые слова: эволюционный сценарий, регуляторный сигнал, частотная матрица, эволюция вдоль дерева, дерево фактора транскрипции.

RECONSTRUCTION OF ANCESTRAL REGULATORY SIGNALS ALONG A TRANSCRIPTION FACTOR TREE, by *K. Yu. Gorbunov*, V. A. Lyubetsky* (Kharkevich Institute for Information Transmission Problems, Russian Academy of Sciences, *e-mail: gorbunov@iitp.ru). A model and an algorithm are proposed to reconstruct ancestral regulatory signals, mostly of protein-DNA interaction, at inner nodes of the transcription factor phylogeny on the basis of contemporary signal distribution. The algorithm also infers an evolutionary scenario, i.e. a set of edges in the tree, along which the signal diverged the most. The model and algorithm were tested with artificial data and biological evidence on signals NrdR, MntR and factor LacI (for the last two cases data are not shown).

Key words: evolutionary scenario, regulatory signal, frequency matrix, evolution along a tree, factor transcription tree.

ПОСТАНОВКА ЗАДАЧИ

Хорошо известны задачи построения дерева эволюции белкового семейства и дерева эволюции семейства видов, а также реконструкции событий молекулярного уровня, имевших место в эволюции белкового семейства [1–4]. Вместе с белковым семейством – фактором регуляции экспрессии гена – эволюционируют сайты связывания этого фактора, поэтому кажется естественной задача реконструировать предковые сайты связывания какого-либо фактора транскрипции. Конкретная задача состоит в том, чтобы вдоль известного дерева эволюции G-фактора транскрипции, используя известные сайты связывания в концевых вершинах этого дерева, реконструировать предковые сайты, т.е. приписать сайты внутренним вершинам этого дерева в качестве предковых. Аналогичный смысл задачи – реконструировать какие-то существенные характеристики расположенных в концевых вершинах сайтов, например, частоты нуклеотидов, встречающихся в них. При этом предполагается, что на некоторых ребрах дерева G происходили "резкие" изменения сайта связывания, например, вследствие изменения фактора транскрипции (точное определение, какое изменение считается "резким", зависит от порога и приводится в следующем разделе). Поэтому в нашу задачу входит и определение ребер, на которых произошли такие "резкие" изменения. Эти ребра будем называть (эволюционно) *значимыми*, а совокупность **ребер** – *носителем* эволюционного сценария. *Эволюционный сценарий* включает, кроме носителя, еще *расстановку* по всем вершинам дерева G реконструированных (предковых) сайтов или их выбранных характеристик, например, матриц частот; определяется именно пара – носитель и расстановка, которые взаимно зависят друг от друга. Эта расстановка возникает при минимизации суммарных изменений по всем ребрам, не вошедшим в носитель сценария, что отражает *принцип максимальной экономии*. Этот принцип, образно говоря, состоит в том, что кроме относительно небольшого числа ребер, на которых действительно произошли эволюционные события, на всех

* Эл. почта: gorbunov@iitp.ru

других ребрах изменения должны быть сколь возможно более "плавными" [5]. Постановка этой задачи иллюстрируется примерами 1–1а в разделе "Результаты работы алгоритма и обсуждение".

Рассмотрим здесь наиболее простой случай белок-ДНКового регуляторного сигнала и соответственно рассмотрим сайты посадки белка-активатора или репрессора. Тогда дано дерево G эволюции соответствующего фактора транскрипции и во всех его концевых вершинах даны наборы сайтов белок-ДНКового взаимодействия, которые выделены в лидерных областях гомологичных генов, находящихся под данной регуляцией. Концевую вершину вместе с *множественным выравниванием* приписанных ей сайтов будем называть *таксоном*, имея в виду, что такой вершине фактически приписывается и набор видов, из которых были выделены эти сайты перед соответствующими генами. В наших данных эти сайты имеют одинаковую длину или они отличаются на единицу, и тогда сайты имеют структуру палиндрома. Если все сайты имеют одинаковую длину, то они записываются друг под другом. Если в одном наборе встречаются палиндромы четной и нечетной длины, то в середину четных палиндромов мы добавляли одну делецию и затем также записывали сайты друг под другом. В обоих случаях в концевой вершине возникает тривиальное множественное выравнивание. Если сайты, приписанные одному таксону, значительно отличаются по длине или мало похожи друг на друга, то рассматривается лишь часть позиций – одинаковое число позиций от каждого сайта; при этом предполагается, что такие существенные позиции в каждом сайте нам известны. В другом случае рассматривается нетривиальное множественное выравнивание этих сайтов, полученное одним из стандартных алгоритмов. Наш алгоритм работает с произвольными множественными выравниваниями, приписанными концевым вершинам-таксонам, но в последнем случае речь обычно идет о мало связанных исходных данных, так что результат счета трудно интерпретировать. Итак, с каждым таксоном сопоставлено свое множественное выравнивание сайтов с n столбцами, где далее n фиксировано. Вместо всего сайта можно рассматривать, в том же смысле, эволюцию нуклеотида в одной фиксированной позиции сайта.

В качестве упомянутой выше характеристики сайта можно рассматривать, например, матрицу позиционных частот множественного выравнивания, свою для каждой концевой вершины; далее обсуждается этот вариант. Матрица позиционных частот – матрица размера $4 \times n$, где четыре строки соответствуют четырем нуклеотидам и n столбцов соответствуют n позициям сайта или, что в нашем случае то же самое, n столбцам множественного выравнивания [6, 7]. При этом столбцы, содержащие делеции, здесь не рассматриваются. Типичной является ситуация, когда имеется несколько эволюционных сценариев, которые ранжируются по *качеству сценария*, определяемому ниже. Наилучшие по качеству эволюционные сценарии сравнивают между собой и с особенностями эволюции самого фактора транскрипции, которая представлена деревом G.

Предлагаемый алгоритм для различных регуляторных сигналов предсказывает расстановки матриц позиционных частот во внутренних вершинах дерева G и эволюционные сценарии на этом дереве. Приводятся результаты тестирования алгоритма на искусственных и биологических примерах.

Итак, пусть для каждой концевой вершины-таксона по приписанному ему множественному выравниванию построены матрицы позиционных частот размера $4 \times n$ с четырьмя строками и n столбцами; смысл чисел строк и столбцов пояснен выше. Сначала рассмотрим один i-й столбец такой матрицы при каждом таксоне, который дает частотное *распределение* четырех нуклеотидов в фиксированной i-й позиции исходного сигнала, для которого была построена эта матрица. Задача состоит в том, чтобы приписать всем предковым вершинам дерева G аналогичные распределения (которые зависят от i, где i фиксировано) наилучшим образом в соответствии с принципом максимальной экономии, который у нашей модели отражается тем способом, при котором некоторая функция F – сумма всех изменений всех распределений по всему дереву G – должна принимать минимальное значение. Расстановка таких распределений во всех внутренних вершинах дерева и для всех значений i, которое меняется от 1 до n, укажет наилучшую расстановку уже самих частотных матриц по внутренним вершинам дерева G. Ребро, на котором происходит "резкое" по некоторому порогу изменение двух распределений, приписанных его концам, входит в эволюционный сценарий, который также зависит от i, будем называть его *i-сценарием*. В алгоритме (см. следующий раздел), естественно возникают несколько i-сценариев и среди них выделяется (обычно один) *наилучший i-сценарий*. Ребра, которые входят в наилучший i-сценарий, назовем *значимыми* по i-й позиции. Предполагается, что на таких ребрах происходили резкие изменения сигнала по i-й позиции, связанные с изменением фактора транскрипции или самого сайта в результате точечных мутаций и т.п.

Носитель *итогового сценария* состоит из ребер, которые входят (иногда с учетом веса) в носители наилучших i-сценариев для *многих значений i*, а сам итоговый сценарий определяется как носитель и соответствующая расстановка матриц по всем вершинам дерева G, полученная из распределений в наилучших i-сценариях при всех i.

При построении итогового сценария можно учесть палиндромную структуру сигнала. Для этого вместо упомянутой выше функции F рассматривается другая функция $\bar{F}$, которая отражает тот факт, что не только сумма всех изменений в распределениях должна быть минимальной, но и эволюция распределений должна идти согласованно в связанных парах i-j позиций палиндрома.

Консервативная позиция сигнала определяется как такая, в которой не все четыре буквы имеют примерно одинаковую частоту (и, значит, одинаковую константу связывания); "примерно одинаковая" означает, конечно, использование некоторого порога (о связи позиционных частот с константами связывания см. [8]). Среди консервативных позиций выделяются *консервативные по одной букве*, когда существенно преобладает одна буква, и аналогично консервативные по двум и трем буквам. В консенсусе консервативные позиции по двум или трем буквам записываются буквами *R* (если *A* или *G*), *Y* (если *C* или *T*) и т.д., что соответствует известному кодированию IUPAC [9]. Консервативные по двум или трем буквам позиции могут допускать много мутаций внутри группы, одинаково приемлемых по этой позиции нуклеотидов. Позиция может быть консервативной на одном этапе эволюции, т.е. в одной связной части дерева G, и неконсервативной на другом этапе [4]. Важно также различать функциональную и эволюционную консервативность.

ОПИСАНИЕ МОДЕЛИ И АЛГОРИТМА

Каждой внутренней вершине v дерева G припишем четыре переменные v_A, v_C, v_G, v_T – значения частот, которые соответствуют нуклеотидам в распределении, отвечающем вершине v. Из принципа максимальной экономии для каждой i-й позиции сигнала отдельно минимизируется функция F, равная сумме *расстояний* между двумя распределениями на концах ребер дерева G; известные значения этих переменных подставляются только в концевых вершинах-таксонах. Накладываются два очевидных *ограничения*: в каждой вершине сумма частот равна 1 и все частоты неотрицательны. Алгоритм допускает и другие ограничения. Для ребра u *обозначим* $F(u)$ слагаемое из суммы F, которое соответствует ребру u.

Опишем общую схему алгоритма, в ходе которой уточняются упомянутые выше понятия, а затем укажем два конкретных *расстояния* (иными словами, метрики) между распределениями, для которых получены приведенные ниже результаты. Модель и алгоритм годятся и для других расстояний.

Минимизируем функцию F при упомянутых линейных ограничениях и полученное решение назовем *промежуточным*. По нему начнем определять i-*сценарий*: рассмотрим *список* ребер u первых по убыванию значений $F(u)$, где длина списка ограничена числовым *параметром*, обозначенным *vet*. В качестве гипотезы принимается, что на этих ребрах произошло эволюционное событие и, следовательно, на них не распространяется принцип максимальной экономии. Такие ребра назовем *значимыми на j-ом шаге* (сейчас $j = 1$). В соответствии с длиной списка возникает *vet* вариантов.

Затем для каждого из j-значимых ребер u выполним то же самое. А именно, минимизируем измененную функцию F: в каждом из этих *vet* вариантов из нее исключается слагаемое $F(u)$, соответствующее текущему j-значимому ребру u, и добавим в i-сценарии по одному новому ребру аналогично тому, как это было сделано выше (шаг $j = 2$). Продолжаем этот процесс, пока число шагов не достигнет значения второго числового *параметра glub*, т.е. последний шаг в работе этой части алгоритма имеет номер $j = glub$. В результате возникает vet^{glub} множеств последовательно исключенных ребер, и каждое такое множество рассматривается как *неупорядоченное* из элементов в числе *glub*. Назовем такое множество *носителем* i-сценария (при данных значениях параметров *vet* и *glub*). Расстановка, которая возникла на последнем $j = glub$ шаге, вместе с соответствующим носителем i-сценария образует сам i-*сценарий*.

Алгоритм постепенно увеличивает параметр *glub*, начиная с нуля. В приведенном ниже счете рассматривали два варианта: фиксированные значения обоих параметров *vet* и *glub*, которые в этом случае явно указываются, и фиксированное значение только *vet* при автоматическом изменении *glub* до достижения *критерия остановки* алгоритма, который указан ниже.

Мощность i-сценария – число элементов в его носителе, равное числу *glub*. Вообще говоря, i-сценарий лучше, если это число меньше.

Итак, возникает некоторое семейство i-сценариев, *качество* каждого из которых оценивается двумя числами. Во-первых, это максимальное значение слагаемого $F(u)$ по всем ребрам u, не входящим в i-сценарий – i-сценарий тем лучше, чем это число меньше. Во-вторых, это сумма значений $\tilde{F}(u)$ функции $\tilde{F}$ по всем ребрам u, не входящим в i-сценарий, где $\tilde{F}(u) = \sum_{d=1}^{4} \sqrt{|u(d, 0) - u(d, 1)|}$ и d пробегает четыре частоты в распределениях $u(d, 0)$, соответствующем началу ребра u, и $u(d, 1)$, которое соответствует концу (к корню) ребра u. i-Сценарий лучше, если это значение меньше.

Второе число будем называть *основным* показателем, а первое – *вспомогательным* показателем. Так же будем называть и списки носителей i-сценариев, упорядоченные по убыванию соответствующих показателей. Для каждого рассмат-

риваемого в ходе алгоритма значения параметра *glub* алгоритм вычисляет основной и вспомогательный списки. Если какой-то носитель *i*-сценария попадает в головные части обоих списков, то он называется наилучшим для фиксированного *i*. Другими словами, если носитель, первый в основном списке, также входит в головную часть вспомогательного списка, то он выделяется как носитель *i-наилучшего* сценария при данных значениях двух параметров *vet* и *glub*.

Алгоритм *останавливается*, если при некотором значении *glub* первая и вторая характеристики качества носителя наилучшего *i*-сценария резко уменьшаются, а до и после этого значения они плавно убывают (в смысле соответствующих порогов). *Показателем пересечения* двух множеств называется отношение мощности их пересечения к мощности их объединения. *Показателем пересечения* нескольких множеств называется среднее арифметическое показателей пересечения по всем парам этих множеств. Если *glub* достигает некоторого порогового значения (например, 10) и указанное выше условие остановки до сих пор не выполнилось, то алгоритм прекращает работу и выдает носитель *i*-сценария, соответствующий значению *glub*, при котором было достигнуто максимальное значение *показателя пересечения* всех носителей наилучших *i*-сценариев для всех просмотренных значений *glub*. Высокий показатель пересечения носителей хотя бы при одном значении *glub* свидетельствует о хорошей обусловленности исходных данных. Если к носителю наилучшего *i*-сценария добавить распредление частот нуклеотидов, которое возникло на шаге работы алгоритма с номером *glub*, то получится ***наилучший i-сценарий***.

В простейшем случае расстояние между распределениями определяется как сумма квадратов разностей соответствующих частот нуклеотидов. Это квадратичная функция с двумя линейными ограничениями, указанными выше, поэтому ее минимизация – задача квадратичного программирования. Такая задача имеет единственное решение. Исключением является случай, когда решение достигается сразу на целом отрезке или многограннике; мы доказали, что для нашей задачи такой случай невозможен. Для задачи квадратичного программирования известны быстрые алгоритмы, работающие даже при многих тысячах переменных и ограничений. При этом алгоритм всегда заканчивает работу и выдает точное решение. При относительно небольшом числе переменных разумно применять более простой алгоритм проекции градиента, который особенно быстро сходится в нашем случае, так как в алгоритме приходится проектировать на пересечение стандартных симплексов. Последнее означает – переменные не повторяются, меняются от 0 до 1 и их сумма равна 1.

Недостаток указанного расстояния в том, что при минимизации большое число небольших изменений сигнала могут оказаться предпочтительнее одного значительного скачка. Это затрудняет поиск *i*-значимых ребер. Простейшее расстояние, для которого этот недостаток не характерен, – расстояние Хелингера, т.е. сумма квадратов разностей квадратных корней из соответствующих частот нуклеотидов. Эта функция обладает и другими достоинствами. Например, штраф за различие двух частот зависит не только от их разности, но и от их отношения: разница между 0.8 и 0.9 штрафуется меньше, чем разница между 0.1 и 0.2. Чтобы избавиться от квадратных корней у функции F с расстоянием Хелингера, новыми переменными объявляются корни из исходных переменных. Тогда функция F снова становится квадратичной, но ограничения-равенства оказываются квадратичными, а не линейными, как это было при первом расстоянии. Новые ограничения будут иметь вид сферических цилиндров, а все допустимое множество – вид пересечения положительных секторов этих цилиндров аналогично случаю первого расстояния.

Потеря линейности ограничений усложнила процесс минимизации, в частности, исчезла гарантированная единственность решения. Однако по-прежнему легко выполняется проектирование на допустимое множество, что сохраняет высокую скорость метода проекции градиента. Нужно только решить вопрос о выборе начальной точки. Мы использовали два приема. Или повторяли процедуру много раз, выбирая начальную точку случайно. Или предварительно решали задачу с первым из указанных расстояний и затем полученное решение использовали как начальную точку при минимизации со вторым расстоянием. Оба варианта дают сходные ответы.

Отмеченный выше недостаток исчезает, если рассматривать следующее третье расстояние: сумму корней из модулей разностей соответствующих частот, т.е. указанную выше функцию $\tilde{F}$, которая, однако, не всюду дифференцируема, и ее минимизация приводит к известным трудностям. Поэтому в нашем алгоритме эта функция применяется косвенно – для оценки качества *i*-сценария.

Согласование наилучших сценариев для различных позиций сигнала. Наилучшие *i*-сценарии при разных значениях *i*, где *i* меняется от 1 до *n*, обычно различаются своими носителями, что вполне естественно: изменения фактора транскрипции и самого сигнала связывания часто затрагивают лишь несколько позиций сигнала, согласованных с его палиндромной структурой. Если событие не привело к радикальному изменению константы связывания фактора с сигналом, то регуляция сохраняется при одновременном измене-

Частоты, принятые в искусственном примере, по листьям исходного дерева "фактора транскрипции"

Таксоны	Распределение в таксоне
1–2	(4) = C: 5/8, A: 2/8, G: 1/8, T: 0
3–16	(2) = T: 5/8, G: 2/8, C: 1/8, A: 0
17–32	(1) = A: 5/8, C: 2/8, G: 1/8, T: 0
33	(5) = T: 5/8, A: 2/8, C: 1/8, G: 0
34–40	(3) = G: 5/8, C: 2/8, T: 1/8, A: 0
41–48	(1) = A: 5/8, C: 2/8, G: 1/8, T: 0
49–50	(6) = G: 5/8, T: 2/8, C: 1/8, A: 0
51–64	(1) = A: 5/8, C: 2/8, G: 1/8, T: 0

нии константы связывания. Если, скажем, две позиции мутировали, то уменьшение константы связывания на одной из них может компенсироваться ее увеличением на другой позиции. В *итоговый эволюционный сценарий*, который уже не зависит от позиции, включаются ребра из разных наилучших *i*-сценариев. А именно, включаются ребра, вес которых выше порога, где *вес* в простейшем случае определяется как число наилучших *i*-сценариев, в которые это ребро входит.

Более полное определение веса ребра учитывает качество наилучшего *i*-сценария, степень консервативности *i*-й позиции, степень значимости данного ребра в *i*-сценарии и также оценку того, насколько часто это ребро одновременно присутствует в связанных между собой позициях, например, в палиндромных парах сигнала. Вес ребра в этом смысле будем называть *полным весом*.

Алгоритм имеет специальный вариант, ориентированный на работу с сигналом, имеющим структуру палиндрома. А именно, фиксируется список всех палиндромных пар позиций *i*,*j*, и алгоритм ищет *наилучшие i*,*j*-*сценарии*, согласованные по комплементарности в позициях *i* и *j*. Такой *i*,*j*-сценарий отражает эволюцию, которая идет с учетом структуры сигнала связывания фактора транскрипции. Для этого определяется новая *палиндромная целевая функция* $\bar{F}$, равная сумме прежних целевых функций F для позиций i и j по отдельности, к которой прибавлена еще сумма штрафов по всем внутренним вершинам v дерева G, где для одной вершины v штраф имеет вид:

$$2\mu[(v_{i,A} - v_{j,T})^2 + (v_{i,G} - v_{j,C})^2 + (v_{i,C} - v_{j,G})^2 + (v_{i,T} - v_{j,A})^2].$$

Таким образом, штрафуется некомплементарность в вершине v, где $v_{i,A}$ – частота буквы A в позиции i для этой вершины и, аналогично, для других переменных $v_{j,T}$, …, $v_{j,A}$. Сравнительная важность комплементарности распределений в связанной паре позиций i, j по сравнению с устойчивостью двух отдельных распределений по позициям i и j регулируется параметром μ в указанной формуле. В приведенных ниже результатах счета принималось $\mu = 1$.

РЕЗУЛЬТАТЫ РАБОТЫ АЛГОРИТМА И ОБСУЖДЕНИЕ

Тестирование алгоритма на искусственном примере

В качестве искусственного дерева G рассмотрим бинарное дерево с 64 концевыми вершинами – таксонами, у которого все пути из корня в таксоны содержат по 6 ребер. Дерево "растет" вниз, нумеруем его таксоны слева направо числами от 1 до 64, а ребра будем обозначать словами из нулей и единиц, которые ведут из корня в конец ребра, где 0 означает "идем налево" и 1 – "идем направо". Для каждого таксона фиксировано одно из следующих распределений по *правилу*, указанному в таблице:

(1) A: 5/8, C: 2/8, G: 1/8, T: 0; (2) T: 5/8, G: 2/8, C: 1/8, A: 0; (3) G: 5/8, C: 2/8, T: 1/8, A: 0;
(4) C: 5/8, A: 2/8, G: 1/8, T: 0; (5) T: 5/8, A: 2/8, C: 1/8, G: 0; (6) G: 5/8, T: 2/8, C: 1/8, A: 0.

В качестве ответа предполагается следующий сценарий с пятью значимыми ребрами: в корне дерева было распределение (1), на ребре 00 оно сменилось на распределение (2), на ребре 100 – на (3), а на ребре 11000 – на (6). В свою очередь, на ребре 00000 распределение (2) сменилось на распределение (4) и на ребре 100000 распределение (3) сменилось на (5). Алгоритм тестировался на многих деревьях и на многих таблицах данных такого сорта. Везде получали результат, подобный описанному ниже.

Итак, результат работы алгоритма таков: при значении параметра *glub* = 1, т.е. среди сценариев с одним значимым ребром, наилучшим в основном списке является сценарий с носителем {100000}; при значении *glub* = 2 – с носителем {00, 00000}; при значении *glub* = 3 – с носителем {00, 100, 11000}, при значении параметра *glub* = 4 – с носителем {00, 00000, 100, 11000}. Видно, что при возрастании параметра *glub* повторяются, в основном, одни и те же ребра, что говорит о хорошей обусловленности начальных данных. Наконец, при *glub* = 5 выдается носитель {00, 100, 11000, 00000, 100000} наилучшего сценария, который и предполагался в качестве ответа, и соответствующая ему расстановка. При *glub* = 5 в пер-

вый раз один и тот же сценарий оказался головным в основном и вспомогательном списках. И впервые на этом сценарии первый и второй показатели качества равнялись нулю, т.е. достигли их минимумов. Для сравнения: при *glub* = 4 у головного сценария в основном списке эти показатели качества равны 0.44 и 9.3, а у головного сценария во вспомогательном списке – 0.08 и 46.1. Указанный выше критерий остановки возрастания параметра *glub* дает значение *glub* = 5.

Для проверки устойчивости алгоритма были проведены испытания со случайным возмущением распределений частот в концевых вершинах-таксонах. Точнее, в каждом таксоне указанного выше дерева каждая из частот нуклеотидов у каждого распределения равновероятно увеличивалась или уменьшалась на число от 0 до 0.1. Наш алгоритм, примененный к таким образом "возмущенным" данным из табл. 1, по-прежнему выдает тот же наилучший сценарий при том же значении *glub* = 5, а распределения в вершинах дерева испытывают небольшие возмущения. Этот сценарий оставался *головным* в основном списке всегда, а во вспомогательном списке – в 80% случаев. В остальных 20% случаев этот сценарий во вспомогательном списке занимал второе место. Это говорит о высокой устойчивости алгоритма к возмущению начальных данных.

Применение алгоритма к биологическим данным

Пример 1. Сигнал NrdR длиной из 16 букв [3] регулирует синтез белков, связанных с репликацией. В качестве дерева *G* сначала было выбрано дерево соответствующих видов с 21-м таксоном, показанное на рис. 1. Пример 1, а также аналогичный результат, касающийся MntR-сигналов, докладывали на конференции BGRS'2006 [10].

Значительное преимущество по числу вхождений в наилучшие *i*-сценарии при разных позициях *i* сигнала получили три ребра, каждое ребро по 11 позиций из 16. Из этих ребер первое ведет к виду *D. radiodurans*, второе – в таксон {*T. maritima*, *T. thermophilus*} и третье – из корня в вершину, помеченную на рис. 1 буквой *a*. Это свидетельствует, во-первых, о том, что NrdR-сигнал для видов *T. maritima* и *T. thermophilus* значительно отличается от такового у других видов. В дереве *G* видов таксон {*T. maritima*, *T. thermophilus*} расположен смежно с корнем, и изменение фактора транскрипции и его сайта для NrdR-регуляции произошло на ребре, ведущем в вершину *a*. Это согласуется с предполагаемым медленным изменением этих видов.

Во-вторых, это свидетельствует о том, что NrdR-сигнал для вида *D. radiodurans* значительно отличается от такового у близких видов. В дереве *G* видов таксон *D. radiodurans* лежит далеко от

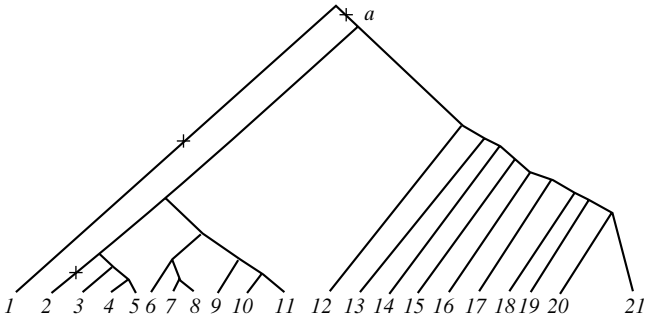

Рис. 1. Дерево *G* видов, из которых извлечен NrdR-сигнал: *1* = {*Thermotoga maritima*, *Thermus thermophilus*}; *2* = {*Deinococcus radiodurans*}; *3* = {*Prochlorococcus marinus*, *Gloeobacter violaceus*, *Synechocystis* sp., *Synechococcus elongates*, *Thermosynechococcus elongates*}; *4* = {*Streptomyces coelicolor*, *S. avermitilis*, *S. scabies*, *Clavibacter michiganensis*, *Leifsonia xyli*, *Corynebacterium* spp., *Mycobacterium* spp.}; *5* = {*Propionibacterium acnes*, *Bifidobacterium longum*, *Thermobifida fusca*}; *6* = {*Staphylococcus aureus*, *S. epidermidis*}; *7* = {*Clostridium acetobutylicum*, *C. tetani*, *C. perfringens*, *C. botulinum*, *C. difficile*, *Thermoanaerobacter tengcongensis*, *Carboxydothermus hydrogenoformans*, *Desulfitobacterium hafniense*}; *8* = {*Bacillus subtilis*, *B. licheniformis*, *B. halodurans*, *B. cereus*, *B. stearothermophilus*}; *9* = {*Enterococcus faecalis*, *E. faecium*}; *10* = {*Streptococcus pyogenes*, *S. agalactiae*, *S. pneumoniae*, *S. mutans*, *Pediococcus pentosaceus*}; *11* = {*Lactobacillus* spp.}; *12* = {*Chlamydia muridarum*, *Chlamydophila pneumoniae*, *Chlamydia trachomatis*, *Chlamydophila abortus*, *C. caviae*, *Treponema denticola*}; *13* = {*Geobacter sulfurreducens*, *G. metallireducens*, *Desulfuromonas acetoxidans*, *Desulfotolea psychrophila*, *Bdelovibrio bacteriovorus*, *Bacteriovorax marinus*, *Myxococcus xanthus*}; *14* = {*Brucella melitensis*, *Mesorhizobium loti*, *Agrobacterium tumefaciens*, *Rhizobium leguminosarum*, *Sinorhizobium meliloti*, *Bradyrhizobium japonicum*, *Rhodopseudomonas palustris*, *Rhodobacter capsulatus*, *Caulobacter crescentus*, *Hyphomonas neptunium*, *Ehrlichia chaffeensis*, *Neorickettsia sennetsu*}; *15* = {*Nitrosomonas eutropha*, *Neisseria meningitidis*, *Methylobacillus flagellatus*, *Ralstonia solanacearum*, *Bordetella pertussis*, *B. bronchiseptica*, *B. avium*, *Burkholderia fungorum*, *Bu. cepacia*, *Bu. pseudomallei*, *Dechloromonas aromatica*}; *16* = {*Xylella fastidiosa*, *Xanthomonas axonopodis*}; *17* = {*Pseudomonas aeruginosa*, *P. putida*, *P. fluorescens*, *P. syringae*}; *18* = {*Vibrio cholerae*, *V. vulnificus*, *V. parahaemolyticus*}; *19* = {*Escherichia coli*, *Salmonella typhi*, *Klebsiella pneumoniae*, *Yersinia pestis*, *Y. enterocolitica*, *Ervinia chrysanthemi*, *E. carotovora*, *Photorhabdus luminescens*}; *20* = {*Pasteurella multocida*}; *21* = {*Haemophilus influenzae*, *H. ducreyi*}.

корня, и можно предположить, что такое изменение характера NrdR-регуляции произошло в процессе образования вида *D. radiodurans* и связано с его быстрой эволюцией.

Пример 2. Теперь рассмотрим дерево белков – фактора транскрипции NrdR, показанное на рис. 2. В качестве дерева *G* выбрано это дерево белков, в котором 31 таксон показан кружочками и некоторые концевые вершины удалены.

В этом дереве значительное преимущество по числу вхождений в наилучшие *i*-сценарии получи-

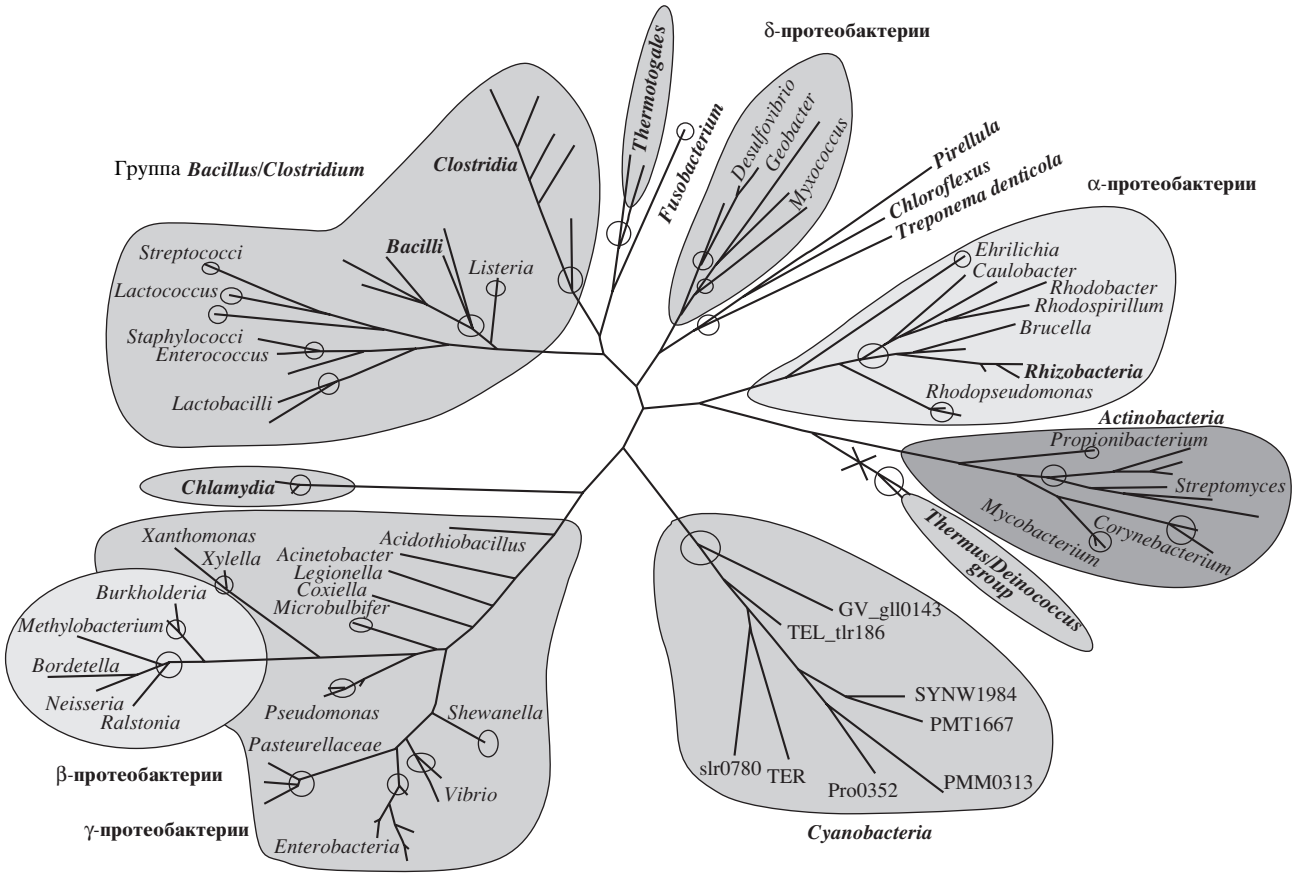

Рис. 2. Дерево *G* таксонов, из которых извлечен NrdR-сигнал. Таксоны выделены кружочками, после каждого таксона указано число сигналов, рассмотренных в нем: *Fusobacterium*, *4*; *Thermotogales*, *4*; *Clostridia*, *25*; *Listeria*, *4*; *Bacilli*, *24*; *Streptococci*, *52*; *Lactococcus*, *6*; *Staphylococci*, *10*; *Enterococcus*, *15*; *Lactobacilli*, *42*; *Chlamidia*, *10*; *Microbulbifer*, *5*; *Xylella*, *Xanthomonas*, *4*; *Burkholderia*, *10*; *Methylobacterium Bordetella*; *Neisseria*, *Ralfstonia*, *10*; *Pseudomonas*, *23*; *Pasteurellaceae*, *13*; *Enterobacteria*, *49*; *Vibrio*, *12*; *Shewanella*, *6*; *Cyanobacteria*, *13*; группа *Thermus/Deinococcus*, *13*; *Mycobacterium*, *20*; *Corynebacterium*, *16*; *Streptomyces*, *12*; *Propionibacterium*, *5*; *Rhodopseudomonas*, *4*; *Rhizobacteria*, *Brucella*, *Rhodospirillum*, *Rhodobacter*, *Caulobacter*, *26*; *Ehrlichia*, *4*; *Treponema denticola*, *Chloroflexus*, *Pirellula*, *4*; *Myxococcus*, *Geobacter*, *8*; *Desulfovibrio*, *14*.

ло одно ребро, ведущее в таксон группу *Thermus*/*Deinococcus*. Это ребро присутствует в 11 сценариях из 16. Оно соответствует трем ребрам, найденным в примере 1. Если принять дерево видов, показанное на рис. 1, без изменений, то не исключено, что произошел горизонтальный перенос гена фактора транскрипции между видами *T. thermophilus* и *D. radiodurans*.

Нами получены сходные результаты (не приводятся) для сигнала MntR с длиной в 22 буквы, который регулирует транспорт марганца, и для соответствующих деревьев видов и факторов транскрипции, а также для факторов из LacI-семейства [11, 12], регулирующих гены катаболизма сахаров, и аналогичного дерева фактора. Во втором случае сайты имеют длину 20.

Переход от консенсуса в одной вершине дерева *G* к консенсусу в другой вершине может происходить путем быстрых компенсаторных замен с сохранением симметричной структуры сайта, например, так происходит в большом LacI-семействе; в этом семействе нами найден случай смены консенсуса, который основан на прохождении через стадию утраты консервативности (данные не приводятся). Алгоритм применялся и к другим семействам сигналов и позволял находить эволюционно значимые ребра, строить эволюционные сценарии по каждой позиции и по парам палиндромно связанных позиций, учитывать структуру сигнала.

Авторы выражают глубокую признательность М. Гельфанду за ценное обсуждение статьи и О. Лайковой, Д. Родионову, А. Селиверстову за помощь и предоставление данных. Особую благодарность авторы выражают А. Витрещаку, важные замечания которого помогли улучшить текст.

СПИСОК ЛИТЕРАТУРЫ

1. Lyubetsky V., Gorbunov K., Rusin L., V'yugin V. 2005. Algorithms to reconstruct evolutionary events at molec-

ular level and infer species phylogeny. In: *Bioinformatics of Genome Regulation and Structure II*. Springer Science a. Business Media, Inc., 189–204.

2. Горбунов К.Ю., Любецкий В.А. 2005. Поиск предковых генов, нарушающих согласованность деревьев белков и видов. *Молекуляр. биология*. **39**, 5, 847–858.
3. Rodionov D.A., Gelfand M.S. 2005. A universal regulatory system of ribonucleotide reductase genes in bacterial genomes. *Trends Genet*. **21**, 385–398
4. Kotelnikova E.A., Makeev V.J., Gelfand M.S. 2005. Evolution of transcription factor DNA binding sites. *Gene*. **347**, 255–263.
5. Fitch W.M. 1971. Towards defining the course of evolution: Minimum change for a specific tree topology. *Syst. Zool*. **20**, 406–416.
6. Schneider T.D. 1986. Information content of binding sites on nucleotide sequences. *J. Mol. Biol*. **188**, 415–431.
7. Stormo G.D., Fields D.S. 1998. Specificity, energy and information in DNA-protein interactions. *Trends Biochem. Sci*. **23**, 109–113.
8. Berg O.G., von Hippel P.H. 1987. Selection of DNA binding sites by regulatory proteins. *J. Mol. Biol*. **193**, 723–750.
9. Интернет-сайт http://www.dna.affrc.go.jp/misc/MPsrch/InfoIUPAC.html.
10. Gorbunov K.Yu., Lyubetsky V.A. 2006. Inferring regulatoty signal profiles and evolutionary events. *Fifth Int. Conf. Bioinformatics Genome Regul. Struct*., thesises. (BGRS'2006). Novosibirsk: IC&G, **3**, 151–154.
11. Laikova O.N. 2002. Systematic prediction of regulatory interactions in the LacI family of transcriptional regulators. *Fifth Int. Conf. Bioinformatics Genome Reg. Struct*., thesises (BGRS'2002). Novosibirsk: IC&G, **2**, 26–28.
12. Gelfand M.S., Laikova O.N. 2003. *Prolegomena to the evolution of transcriptional regulation in bacterial genomes in frontiers in computational genomics*. Wymondham, UK: Caister Academic Press.