Зверков О.А.

Институт проблем передачи информации им. А.А. Харкевича Российской академии наук (ИППИ РАН), научный сотрудник, zverkov@iitp.ru

Использование быстрых алгоритмов в задаче кластеризации последовательностей

КЛЮЧЕВЫЕ СЛОВА:

Кластеризация, быстрый алгоритм, биоинформатика.

АННОТАЦИЯ:

Для современной молекулярной биологии характерно экспоненциальное увеличение объёмов доступных геномных данных. Несмотря на постоянный рост производительности вычислительной техники, одного этого далеко не достаточно для удовлетворения растущих потребностей биоинформатики. Поэтому создание и использование быстрых вычислительных алгоритмов в задачах биоинформатики сохраняет и всё увеличивает свою актуальность. В данной работе приводится пример использования таких алгоритмов в задаче кластеризации белков.

Введение

Одной из важнейших задач биоинформатики является кластеризация генов или белков (будем для определённости говорить о белках) из различных видов живых организмов (ниже слово «вид» употребляется везде только в этом специальном смысле), т.е. разделение данного множества белков на кластеры (непересекающиеся подмножества) таким образом, что (неформально говоря) внутри одного кластера находятся гомологичные (родственные) белки из разных видов, а в разных кластерах — не гомологичные (не родственные или отдалённо родственные) белки. Более подробно задача кластеризации белков обсуждается, например, в нашей статье [1]. Одной из возможных формализаций этой задачи является оригинальный алгоритм, описанный в той же статье и коротко изложенный для удобства в следующем пункте.

Алгоритм кластеризации белков

Математически решается следующая задача. Дано множество белков (последовательностей в двадцатибуквенном алфавите), из набора видов живых организмов. Требуется построить кластеризацию (т.е. разбиение этого множества белков на попарно непересекающиеся подмножества), так чтобы в один кластер попали сходные по последовательности белки из разных видов, а белки из одного вида как можно реже попадали в один кластер. Нами предложен следующий алгоритм нахождения упомянутых

кластеров, результаты которого согласуются с биологическими наблюдениями. Кластеры формируются измельчением, начиная с единственного кластера, содержащего все белки. Общий план работы алгоритма показан на рис. 1.

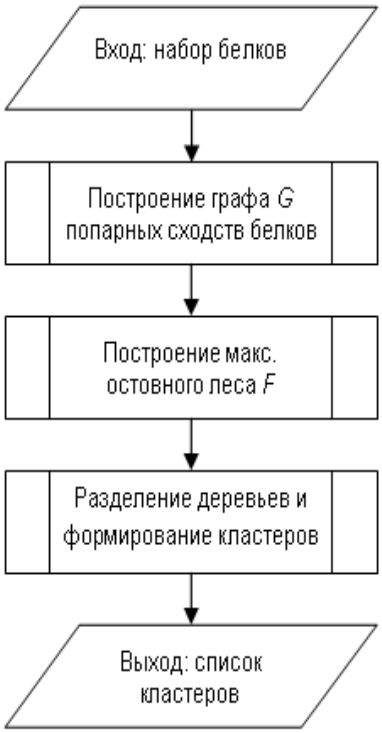

Рисунок 1. Общий план алгоритма кластеризации

Пусть задан набор видов S_i и для каждого вида перечислены его белки P_{ij}. Для всех пар белков (P_{ij}, P_{kl}) из всех пар видов вычисляется значение сходства $s_0(P_{ij}, P_{kl})$ белков как качество оптимального глобального выравнивания этих последовательностей. Затем алгоритм вычисляет значение *нормированного сходства* $s(P_{ij}, P_{kl})$ белков: $2\,s_0(P_{ij}, P_{kl})(s_0(P_{ij}, P_{ij}) + s_0(P_{kl}, P_{kl}))^{-1}$.

Рассматривается *полный* неориентированный граф G_0 с множеством вершин $\{P_{ij}\}$, в котором каждому ребру (P_{ij}, P_{kl}) приписано значение $s(P_{ij}, P_{kl})$, которое будем называть *весом* этого ребра. По G_0 строится разреженный граф G, включающий лишь рёбра (P_{ij}, P_{kl}), удовлетворяющие условиям:

$$s(P_{ij}, P_{kl}) = \max_m s(P_{im}, P_{kl}) = \max_m s(P_{ij}, P_{km}) \text{ и } s(P_{ij}, P_{kl}) \geq L,$$

где максимумы берутся по всем белкам из соответствующих видов, *i*-го и *k*-го, а L — параметр алгоритма, по умолчанию равный нулю. Если $i = k$, то предполагается ещё условие $m \neq l$.

В полученном графе *G* алгоритм процедурой Крускала строит максимальный (по суммарному весу рёбер) остовный лес *F* (рис. 2). А именно, в *G* перебираются рёбра в порядке убывания их веса (при совпадении весов сначала выбираются рёбра, соединяющие белки одного

вида), которые объявляются рёбрами строящегося леса *F*, если добавление к *F* очередного ребра из *G* не приводит к появлению в *F* цикла. В результате *F* не содержит циклов, т.е. является лесом, и включает все вершины из *G*.

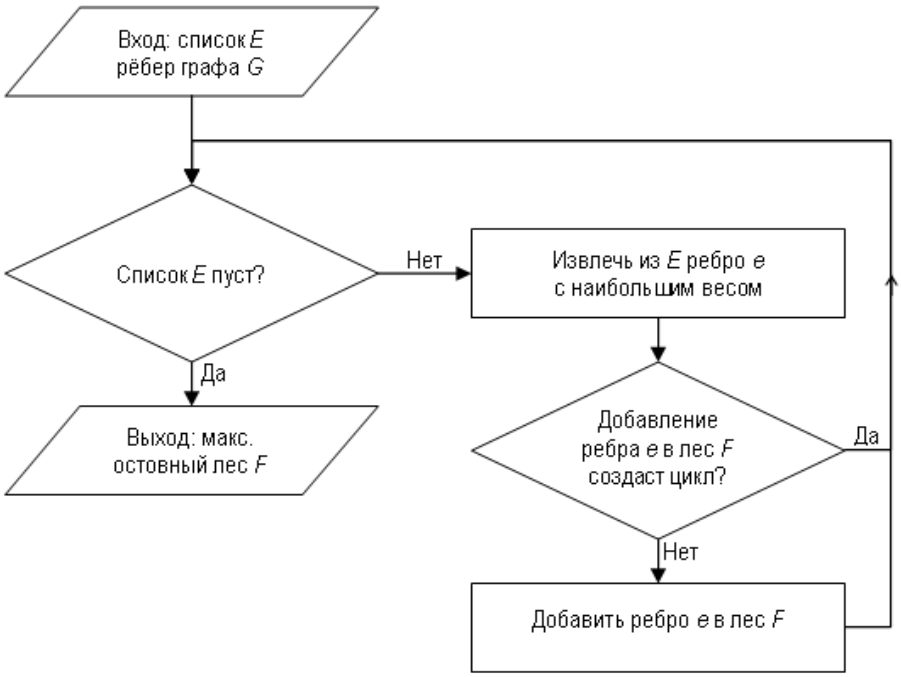

Рисунок 2. Схема алгоритма построения максимального остовного леса. *В начале список E содержит все рёбра графа G, а лес F — все вершины графа G. В результате: список E пуст, а лес F накрывает все вершины графа G и его вес максимальный.*

Затем к лесу *F* применяется следующая процедура разделения деревьев (рис. 3), строящая набор *C* искомых белковых кластеров. Пусть *T* — дерево из *F* и e_0 — ребро в *T* с минимальным по всем ребрам в *T* весом s_0. Если $s_0 < H$, где *H* — параметр алгоритма, и *T* не удовлетворяет сформулированному ниже *критерию сохранения дерева*, то *T* заменяется в *F* на два новых дерева *T′* и *T″* путём удаления из *T* ребра e_0; в противном случае (т.е. критерий выполнен или $s_0 \geq H$) дерево *T* перемещается из *F* в список *C*.

Критерий сохранения дерева *T* состоит в выполнении трёх условий (рис. 4):

(1) $|T| \leq pn$, где $|T|$ — число вершин в дереве *T*, *n* — число всех видов в исходном наборе, *p* — параметр алгоритма;

(2) ребро (P_{mq}, P_{kl}) с минимальным в *T* весом соединяет белки P_{mq} и P_{kl}, у которых $m \neq k$;

(3) любая пара вершин P_{mq} и P_{ml} дерева *T*, соответствующих белкам из одного вида, соединена в *T* путём, состоящим из вершин, соответствующих белкам того же вида (т.е. подграфы, которые состоят из вершин, относящихся к одного вида, связны).

Если в *F* ещё остались деревья, то рассматривается следующее дерево *T* из *F*, иначе алгоритм завершает работу. Полученный в результате набор деревьев *C* представляет собой кластеры исходных белков: один кластер

состоит из последовательностей, приписанных всем вершинам одного дерева.

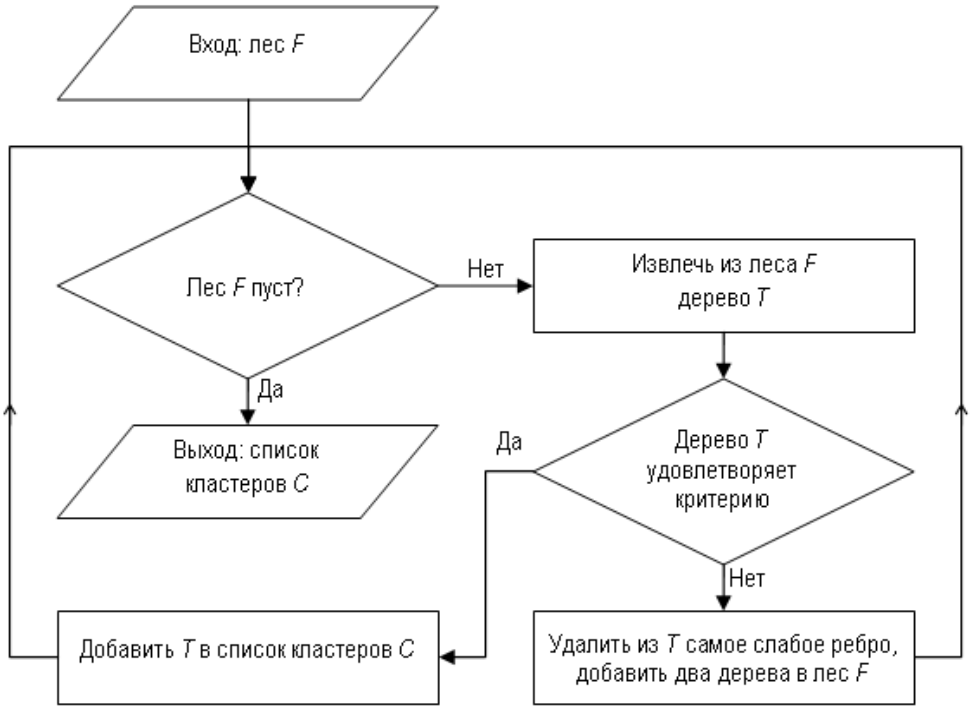

Рисунок 3. Схема алгоритма разделения леса и формирования кластеров. *Вначале лес F — максимальный остовный лес G, а список кластеров C пуст. В результате лес F пуст, а список C содержит набор искомых кластеров*

Технический анализ

Как видно из описания алгоритма, для его работы требуется вычислить значения сходства для всех пар белков из данного набора. При характерных объёмах протеома (набора различных белков одного организма) порядка 10^4 белков, для кластеризации белков нескольких десятков организмов требуется порядка 10^{11} операций построения оптимального парного выравнивания белков. (Для наглядности: эта величина порядка числа секунд в тысячелетии.) В свою очередь, нахождение оптимального парного выравнивания требует выбора одного из огромного числа вариантов взаимного расположения двух аминокислотных последовательностей. Число возможных вариантов выравнивания пары белков экспоненциально зависит от их длины, поэтому попытка найти оптимальное выравнивание путём построения всевозможных выравниваний и выбора из них лучшего требует невыполнимого на практике за разумное время числа операций даже для относительно коротких белков.

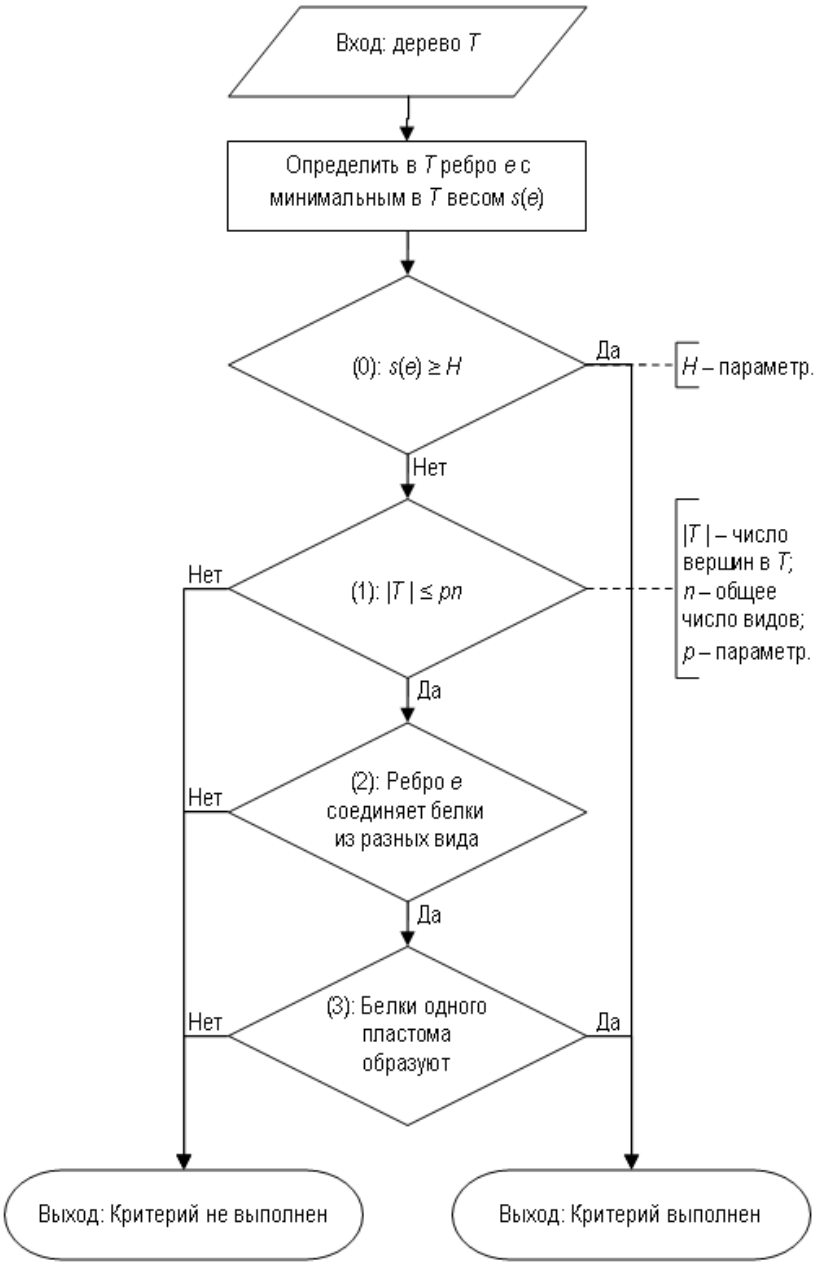

Рисунок 4. Схема проверки критерия сохранения дерева

Для решения этой задачи используется алгоритм [2], основанный на общем подходе, известном как *метод динамического программирования*. Его применение основано на наблюдении, что при фиксированном выравнивании некоторых префиксов двух данных последовательностей, выравнивание можно продолжить на следующую позицию лишь тремя способами (a/b, a/–, –/b, где a —очередная буква первой последовательности, b — очередная буква второй последовательности, “–” — символ делеции). То есть, если ранее найдено оптимальное выравнивание для некоторых префиксов данных последовательностей, его можно оптимальным образом продолжить на одну позицию, выбрав из трёх упомянутых вариантов. Такой подход позволяет снизить вычислительную

сложность построения оптимального выравнивания до квадратичной, сведя его (в случае линейного штрафа за делеции) к $m \times n$ операциям вычисления максимума трёх целых чисел (где m и n — длины последовательностей).

Реализация указанного подхода на низкоуровневом языке программирования позволяет достигнуть на современных микропроцессорах времени построения выравнивания двух белков типичных размеров порядка нескольких миллисекунд. При использовании параллельных вычислений на высокопроизводительном компьютерном кластере с числом параллельных потоков вычислений порядка тысячи, удаётся выполнить порядка 10^{10} выравниваний в сутки, что позволяет произвести все необходимые для кластеризации операции сравнения белков за приемлемое время.

Вторая техническая сложность — хранение и обработка больших объёмов данных о сходстве белков. Как указано выше, число таких значений имеет порядок 10^{11} (т.е. порядка терабайта данных), что делает оперативный поиск нужного значения технически сложной задачей. Для её решения используется описанный выше приём разрежения графа. После вычисления всех значений сходства для пары белков из двух фиксированных геномов можно сразу найти для каждого белка из первого генома максимально сходные белки из второго генома и в дальнейшем хранить и использовать лишь значения сходства для взаимно максимально сходных белков. Это позволяет снизить объём используемых данных примерно на четыре порядка (характерное число белков одного протеома).

После удаления несущественных рёбер (разрежения графа) остаётся порядка 10^7 рёбер. Для дальнейшего удаления избыточных рёбер в алгоритме используется метод Крускала [3] построения максимального (по весу) остовного леса, в быстрой реализации которого существенную роль играет хранение компонент связности в виде *системы непересекающихся множеств*. Это позволяет выполнять разделение графа на компоненты связности и построение в каждой из них максимального остовного дерева за время порядка $O(E \log(E)) + O(E\ \alpha(E, V))$, где α — функция, обратная к функции Аккермана (которую можно оценить константой для всех практических задач). Тот же быстрый алгоритм используется и на стадии разделения деревьев, что существенно для производительности, так как эта процедура повторяется при каждом удаления ребра.

Примеры результатов

Описанный алгоритм был успешно применён для построения кластеризации кодируемых в пластидах и митохондриях белков широких таксономических групп растений и простейших; в частности, в пластомах 186-ти видов цветковых растений; в митохондриях 66-ти видов таксономической группы зелёных растений (Viridiplantae) [1]; белков родофитной и хлорофитной (водоросли и мохообразные) ветвей пластид [4; 5]. На этой основе получены биологические результаты. Например,

найдены белковые семейства, специфичные для пластомов небольших таксономических групп водорослей и простейших [4]; проведён поиск и анализ РНК-полимераз в ядерных геномах споровиков [4]; изучены вставки прямых повторов в микроэволюции митохондрий и пластид растений [5]; в митохондриях винограда (*Vitis vinifera*) найдены уникальные для них белки, которые типичны для пластид, что позволяет предсказать горизонтальный перенос из пластид в митохондрии [1]. В настоящее время алгоритм применяется для кластеризации полных протеомов нескольких десятков животных, что соответствует масштабам, используемым для примера выше.

Литература

1. Любецкий В.А., Селиверстов А.В., Зверков О.А. Построение разделяющих паралоги семейств гомологичных белков, кодируемых в пластидах цветковых растений // *Математическая биология и биоинформатика*. 2013. Т. 8, № 1. С. 225–233.
2. Needleman S.B., Wunsch C.D. A general method applicable to the search for similarities in the amino acid sequence of two proteins // *Journal of Molecular Biology*. 1970. V. 48, No. 3. P. 443–453.
3. Kruskal J.B. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem // *Proc. AMS*. 1956. V. 7, No. 1. P. 48–50.
4. Зверков О.А., Селиверстов А.В., Любецкий В.А. Белковые семейства, специфичные для пластомов небольших таксономических групп водорослей и простейших // *Молекулярная биология*. 2012. Т. 46, № 5. С. 799–809.
5. Зверков О.А., Русин Л.Ю., Селиверстов А.В., Любецкий В.А. Изучение вставок прямых повторов в микроэволюции митохондрий и пластид растений на основе кластеризации белков // *Вестник Московского университета. Серия 16: Биология*. 2013. № 1. С. 8–13.