ДОКЛАДЫ АН ВШ РФ
2016 октябрь–декабрь № 4 (33)

ТЕХНИЧЕСКИЕ НАУКИ

УДК 004.421

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ ФИЛЬТРАЦИИ ПОВТОРОВ В ДАННЫХ NGS ILLUMINA*

А.Н. Цыбин[1], В.В. Шаров[1], Ю.А. Путинцева[2], С.И. Феранчук[1,3], Д.А. Кузьмин[1]
[1]*Сибирский Федеральный университет*
[2]*Институт леса им. В.Н. Сукачева СО РАН*
[3]*Иркутский национальный исследовательский технический университет*

В статье рассматривается подход предобработки фрагментов (ридов), полученных по NGS технологии, позволяющий значительно сократить объем входных данных, используемых в сборке больших геномов. Основная идея – фильтрация ридов от повторяющихся элементов, не используемых в белковом анализе данных. Разработан параллельный вероятностный алгоритм фильтрации, позволяющий значительно сократить результирующее время de Novo сборки генома с минимальной потерей кодирующей информации. Реализация алгоритма направлена на достижение максимального быстродействия. Корректность работы алгоритма и программы тестировалась на модельном растении Arabidopsis Thaliana [6], чья длина генома составляет около 140 млн пар нуклеотидных оснований (п.н.о.). Сборка генома осуществлялась геномным ассемблером SPAdes. Верификация проводилась методом выравнивания ридов РНК на полученную сборку. В результате работы программы достигнуто значительное (более 20 %) сокращение исходных данных NGS с потерей кодирующей информации в пределах 0,005 %, при уменьшении времени работы геномного ассемблера SPAdes более чем в 2 раза.

Ключевые слова: параллельный алгоритм, кластеризация, биоинформатика, повторы, фильтрация, ассемблирование генома, Illumina, SPAdes, Abyss.

DOI: 10.17212/1727-2769-2016-4-99-110

Введение

Ассемблирование геномов является важным этапом исследования организмов методами молекулярной биологии. Ассемблирование выполняется с помощью специальных пакетов программ на основе результатов обработки биологического материала с помощью специальных устройств, называемых секвенаторами. В последнее время произошел технологический прорыв в методах определения последовательностей ДНК в биоматериале. Устройства, называемые секвенаторами нового поколения (NGS), выдают в результате своей работе очень большое количество коротких фрагментов нуклеотидов, называемых ридами (прочтениями). На основании этих данных можно с определенной точностью получить последовательность генома для исследуемого организма, в результате процесса ассемблирования ридов. Среди алгоритмов ассемблирования различают ассемблирование на основании сходства с уже известным геномом родственного вида, либо ассемблирование без привлечения дополнительной информации, которое называется ассемблированием de Novo.

Для геномов хвойных, имеющих значительный размер, который составляет от 12 до 30 Gb и содержащих до 82 % повторяющихся элементов (повторов), de

Исследование выполнено в рамках проекта «Геномные исследования основных бореальных лесообразующих хвойных видов и их наиболее опасных патогенов в Российской Федерации», финансируемого Правительством РФ (договор № 14.Y26.31.0004).

Novo сборка (ассемблирование) является достаточно сложным процессом, требующим значительных вычислительных ресурсов [1] и подходов, отличных от подходов, применяемых в сборке небольших геномов. Собственно основными проблемами являются большой объем входных данных и наличие высокого процента повторяющихся элементов, что усложняет сборку программами, использующими методы [2], основанные на графах де Брёйна. Многие современные ассемблеры, такие как SPAdes [3] или Abyss [4], включают в себя специальные процедуры для учета повторов, но не способны обработать большие объемы входных данных, требуемые для сборки геномов хвойных. Соответственно, актуальной является задача разработки методов подготовки входных данных (ридов, полученных по технологии NGS) ещё до этапа ассемблирования, позволяющих уменьшить их объем без потери кодирующей информации и, как результат, значительно упростить этап de Novo ассемблирования больших геномов.

1. Основные понятия

Геном (англ. genome) – уникальная информация, характеризующая организм, извлекаемая из ДНК и представленная как строки в алфавите {A, C, G, T}.

Секвенирование (англ. sequencing) – общее название методов, которые позволяют установить последовательность нуклеотидов в молекуле ДНК или РНК.

Next-Generation sequencing (NGS) Illumina – технология секвенирования нового поколения от компании Illumina.

Рид (англ. read) – отдельная последовательность (фрагмент ДНК), полученная в результате секвенирования, имеет длину во много раз меньше, чем сам геном.

Контиги – набор фрагментов ДНК, которые в совокупности представляют собой консенсусную область ДНК.

Повторы – часто повторяющиеся последовательности в геноме.

Ассемблирование (сборка) генома – процесс объединения ридов (геномных данных) в продолжительные последовательности генома (контиги).

de Novo сборка – ассемблирование генома впервые, без использования уже существующих сборок.

Кодирующие данные – последовательность нуклеотидов в геноме, кодирующая белки.

2. Алгоритм

2.1. Идея

В процессе секвенирования повтор, как и любая другая часть генома, разбивается на риды определенной длины. Существует два возможных сценария такого дробления: когда длина рида больше (или равно) длины повтора и когда меньше. В том случае, когда длина рида оказывается больше длины повтора, количество ридов, содержащих повтор, будет приблизительно равняться изначальному количеству встреч этого повтора в геноме. Во втором случае, когда длина рида меньше длины повтора, риды могут разбиваться на две категории:

- **риды-повторы** – риды, являющиеся подпоследовательностью какого-либо повтора;
- **концевые риды** – риды, в которых представлены лишь начало или конец повтора, а также часть генома, не относящаяся к повтору.

На рис. 1 повтор R разбит на риды, составляющих группы A и B. Группа A состоит из концевых ридов, а группа B – из ридов-повторов.

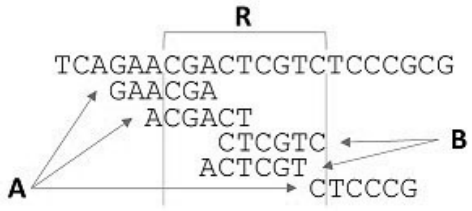

Рис. 1 – Две категории ридов повтора

Fig. 1 – Two categories of reads of repeat

Наиболее интересна категория ридов-повторов, которые в отличие от концевых ридов состоят исключительно из частей повтора. Для этой категории, исходя из положения, что повторы встречаются в геноме более одного раза, можно ввести следующие определения:

• **два рида называются «похожими»**, если они имеют N и более общих подпоследовательностей фиксированной длины (далее, к-меров). На рис. 2 представлены два таких рида, являющиеся похожими при трех одинаковых к-мерах длины 9;

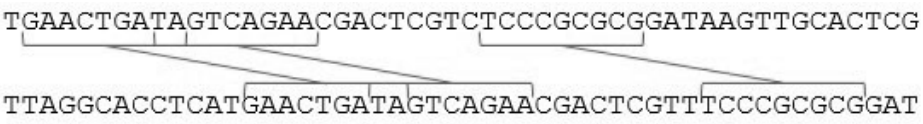

Рис. 2 – Похожие риды

Fig. 2 – Similar reads

• **кластер ридов считается кластером с ридами-повторами**, если все риды в нём похожи между собой и размер кластера больше или равен заданному порогу. На рис. 3 представлен пример подобного кластера (для удобства риды выравнены по подпоследовательности «CCCGCGCGG»).

```
----TGATAGTCAGAACGACTCGTTCCCGCGCGGATAAGTTACACTCGGAAGTC-------
----------GACAGAACGACTTTTCCCGCGCGGATAAGTTACACTCGGACGTCTGTGTT-
GAACTGATAGTCAGAACGACTGTTTCCCGCGCGGATAAGTTACACTCGGA-----------
----TGATAGTCAGAACGACTCTTTCCCGCGCGGATAAGTTACACTCGGACGTC-------
---------GTCAGAACGACTGTTTCCCGCGCGGATAAGTTACACTCGGACGTCTGTGT--
----------TCAGAACGACTGTTTCCCGCGCGGATAAGTTACACTCGGACGTCTGTGTT-
------ATAGTCAGAACGACTGTTTCCCGCGCGGATAAGTTACACTCGGACGTCTG-----
-----GATAGTCAGAACGACTCGTTCCCGCGCGGATAAGTTACACTCGGAAGTCT------
-------TAGTCAGAACGACTGTTTCCCGCGCGGATAAGTTACACTCGGACGTCTGT----
GAACTGATAGTCAGAACGACTCGTTCCCGCGCGGATAAGTTACACTCGGA-----------
-GAACTGATAGACAGAACGACTTTTCCCGCGCGGATAAGTTACACTCGGAC----------
-----ATAGTCAGAACGACTCCTTTCCCGCGCGGATAAGTTACACTCGGACGTCA------
-----------CAGAACGACTCGTTCCCGCGCGGATAAGTTACACTCGGACGTCTGTGTTT
```

Рис. 3 – Кластер ридов-повторов

Fig. 3 – Cluster of reads-repeats

«Кластером уникальных ридов» является кластер, размер которого меньше некоторого заданного порога. Далее по тексту такие риды именуются «уникальными

ридами» и рассматриваются вне контекста кластеров. Таким образом, кластеризовав риды по похожести между собой и отфильтровав кластеры с ридами-повторами, мы получаем лишь уникальные риды.

2.2. Основные этапы работы алгоритма

Этап предобработки. Рабочими единицами алгоритма являются к-меры, используемые главным образом для определения похожести ридов, поэтому предварительно разбиваем все риды на к-меры заданной константной длины. Сравнивание к-меров ридов, вместо самих ридов, позволяет уменьшить влияние мутаций, вставок, удалений или ошибок чтения нуклеотидов секвенатора. Каждый к-мер при генерации соответствует какому-либо риду. Эта информация сохраняется в структуре данных, где ключом является к-мер, а значение – вектором индексов ридов, в которых данный к-мер встречается (далее такая структура называется «структура индексации ридов по к-мерам» или «СИРК»).

На рис. 4 показана графическая интерпретация СИРК. Рид «ATGCGA» под номером 4 разбивается на 4 к-мера. В СИРК заносится информация о том, что к-меры найдены в риде. Все массивы индексов хранятся в отсортированном виде, поскольку разбиение происходит последовательно.

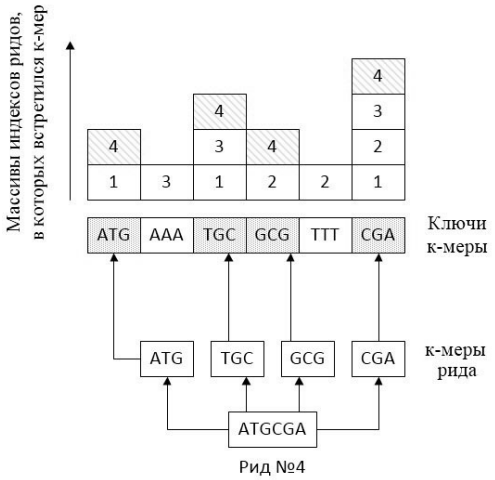

Рис. 4 – Графическое представление СИРК

Fig. 4 – Graphical representation for SIRK

Этап кластеризации. Следом за этапом предобработки начинается этап кластеризации ридов, необходимый для определения уникальных ридов и ридов-повторов. Каждый рид рассматривается отдельно, как совокупность к-меров его составляющих, где из к-меров рида случайно выбирается заданное их количество. Выбранные к-меры рассматриваются как ключи к структуре СИРК для доступа к векторам индексов ридов, где эти к-меры были найдены. Находим бинарное пересечение этих векторов, результирующий вектор будет хранить в себе индексы ридов, где встретились данные к-меры, т. е. индексы ридов, похожих на рассматриваемый рид и образующих кластер.

На рис. 5 дан пример поиска ридов, похожих на «ATGCGA» (для наглядности, показаны только используемые массивы индексов). Из рида случайным образом выбирается заданное количество к-меров (на рисунке – 3 к-мера), в СИРК по этим к-мерам находятся массивы с индексами ридов, где встречаются данные к-меры.

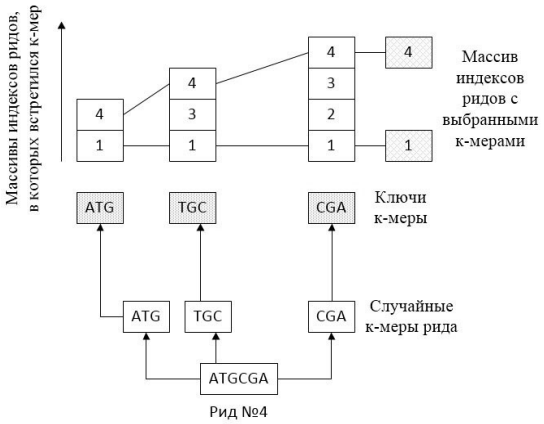

Рис. 5 – Использование СИРК

Fig. 5 – Usage of SIRK

Применив к ним бинарное пересечение, получаем вектор похожих с «ATGCGA» ридов (номера 1 и 4), являющийся кластером похожих ридов.

После того как кластер создан, определяется его тип. Если размер кластера больше заданного порога, то кластер является ридом-повтором, иначе – полагаем, что это кластер уникальных ридов. В итоге мы получаем уникальные риды, являющиеся результатом фильтрации исходных ридов от ридов-повторов.

2.3. Дополнения

Оптимальная длина ридов. Алгоритм основан на фильтрации ридов, полученных по NGS технологии, и имеет ограничение на их длину. Оптимальными для данного алгоритма являются риды длиной 100 п.н.о.

Похожесть ридов. Параметры похожести ридов позволяют задавать качество отбора ридов-повторов, что может значительно ускорить или замедлить работу программы. При этом неправильно подобранные параметры могут привести к появлению некорректных результатов. Например, при чрезмерном уменьшении количества общих к-меров у похожих ридов или при уменьшении длины к-мера похожими могут быть определены риды, не являющиеся в действительности таковыми.

На рис. 6 приведен пример неверного определения похожести ридов из-за задания слишком малой длины к-мера.

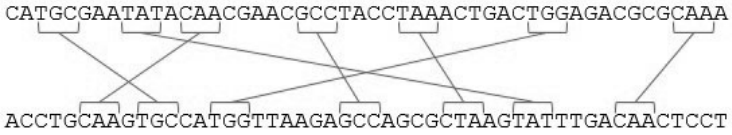

Рис. 6 – Неверное определение похожих ридов

Fig. 6 – Wrong determination of similar reads

Случайный выбор к-меров. При случайном выборе к-меров для определения похожести ридов могут возникнуть следующие два ложных результата:

• ложноположительный результат – свидетельствует о том, что риды похожи, когда в действительности они не являются ни концевыми ридами повтора, ни ридами-повторами. На рис. 7 видно, что к-меры, выбранные случайно, попали

в общую одинаковую подпоследовательность ридов, в то время как оставшиеся части ридов не похожи между собой;

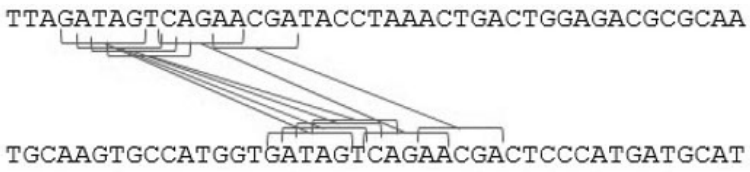

Рис. 7 – Ложноположительный результат определения похожести ридов

Fig. 7 – False-positive determination of the reads similarity

• ложноотрицательный результат – ложное свидетельство о том, что риды не похожи. На рис. 8 в один из выбранных к-меров попал нуклеотид, мутировавший или неверно прочитанный во втором риде.

Рис. 8 – Ложноотрицательный результат определения похожести ридов

Fig. 8 – False-negative determination of the reads similarity

Для уменьшения появления ложных результатов при случайном выборе к-меров добавлен такой параметр, как «шаг выборки к-меров». На этапе кластеризации при повторном дроблении рида на к-меры каждый последующий к-мер выбирается на расстоянии «шага» от предыдущего. На рис. 9 приведен пример дробления рида на к-меры с шагами 1 и 3.

Рис. 9 – Дробление рида на к-меры с разным шагом

Fig. 9 – Splitting of the read with various steps

При таком подходе количество общих к-меров для определения похожести ридов сокращается, но длина общей подпоследовательности ридов увеличивается, что приводит к уменьшению ложноположительных результатов.

В случае с ложноотрицательным результатом использование «шага выборки к-меров» приводит к уменьшению количества к-меров, включающих в себя мутацию, вставку, удаление или ошибку. Так, на рис. 10 количество к-меров, содержащих мутацию, равно пяти при шаге, равном единице и двум – при шаге, равном 3.

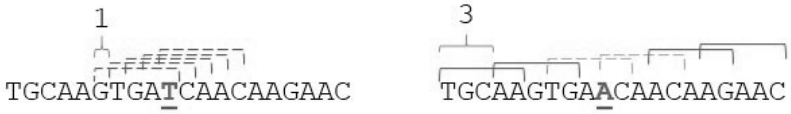

Рис. 10 – Содержащие ошибку к-меры при разных шагах

Fig. 10 – K-mers with errors with various steps

Следует учитывать, что ложноотрицательный результат менее критичен, чем ложноположительный, поскольку ложноположительный результат приводит к образованию кластера из непохожих ридов, т. е., происходит фильтрация уникальных ридов. Ложноотрицательный же результат, наоборот, не приводит к образованию кластера, а приводит к определению повторов как уникальных ридов.

Предположим, что в исходных данных 100 ридов потенциально могут образовать кластер, так как являются частями одного повтора. Алгоритм, проверяя один из ридов этого кластера, раздробил его на к-меры и случайно выбрал часть из них, при этом один из к-меров был выбран с мутацией. В результате рид не образовал кластер ридов-повторов, хотя все 100 ридов могут быть похожи между собой при ином выборе к-меров.

После проверки первого рида алгоритм через некоторое время проверит 2-й рид из этого (еще не сформированного) кластера. Если выбор к-меров опять окажется неподходящим, выбирается 3-й рид из кластера и так далее. В результате будет проведено 100 попыток определить кластер ридов-повторов. Это значит, что чем выше порог размера кластера ридов-повторов или чем чаще встречается повтор в геноме, тем меньше вероятность ложноотрицательного результата.

2.4. Оптимизация алгоритма

Вычислительная сложность алгоритма складывается из двух показателей: объема занимаемой памяти и скорости выполнения:

- скорость выполнения по большей части зависит от этапа построения кластеров похожих ридов. Предобработка же занимает лишь малую часть времени выполнения при использовании оптимальных параметров, приведенных далее;
- расход памяти зависит от двух показателей: объема исходных данных умноженного на 5 (сюда включены расходы на дополнительные структуры данных и саму СИРК) и количества фильтруемых данных.

Предпринятые шаги по оптимизации скорости выполнения:

- сравнение строк – медленная операция, поэтому по возможности необходимо от нее избавиться. Так, к-меры можно представить в виде целых чисел (хешей к-меров), кодируя один нуклеотид как 2 бита информации и далее оперировать уже целочисленными данными;
- учитывая частое обращение к структуре СИРК, выгодно вместо «словаря» (структуры типа «ключ-значение») использовать обычный массив, где в качестве индекса служит хэш к-мера, а размер массива равен всем возможным вариантам к-мера заданной длины. Это накладывает ограничение на длину к-мера, но позволяет значительно сократить время расчетов;
- алгоритм обладает хорошими свойствами параллелизма – каждый отдельный рид на этапе кластеризации может быть проверен отдельным потоком.

Параллельное построение СИРК

В последовательной реализации алгоритма построения СИРК все риды анализируются последовательно, соответственно их индексы заносятся в СИРК по порядку, в результате СИРК автоматически приобретает важное для бинарного пересечения свойство: любой его столбец содержит номера ридов в отсортированном виде.

В параллельной реализации каждому потоку выделяется по последовательному блоку ридов в объеме, достаточном для того, чтобы за один раз все потоки

обработали все риды. В результате выполнения каждого потока формируется часть СИРК, содержащая лишь индексы ридов, за которые был ответственен данный поток.

Для получения полной СИРК необходимо объединить все ее части в порядке следования блоков ридов – таким образом, столбцы СИРК по-прежнему будут отсортированы по умолчанию.

Выполнение промежуточных этапов слияния по две части СИРК, как при классическом подходе «разделяй и властвуй», не приведет к ускорению из-за дорогой операции копирования больших объемов данных.

Параллельное нахождение похожих ридов

Как уже было сказано, на этапе нахождения похожих ридов каждый рид может обрабатываться отдельным потоком, так как после построения СИРК больше не изменяется. При параллельной реализации данного этапа необходимо разграничивать доступ к массиву флагов для ридов, указывающих на то, принадлежит ли рид какому-либо кластеру или нет.

3. Тестирование

Тестируемые исходные данные представлены в формате fasta [5]. Во всех случаях при указании размера исходных данных подразумевается размер файла ридов с их описанием. Пример описания и их длина приведены в табл. 1.

Таблица 1 / Table 1

Пример описания ридов
Example of reads definition

Организм / Organism	Описание / Definition	Длина / Length
Arabidopsis Thaliana	>SRR492411.3 LAMARCK:3111:C01ULACXX:1:1101:1430:2170 length=101	63

Характеристики вычислительной системы, используемой при экспериментах: объем ОЗУ – 3 ТБ, количество ядер – 96.

При реализации параллельного алгоритма использовался класс thread, стандарт C++ 11.

Алгоритм тестировался на модельном растении Arabidopsis Thaliana [6], чья длина генома составляет около 140 млн п.н.о.

Для настройки фильтрации проводился ряд экспериментов, в которых изменялись следующие параметры: длина к-мера, количество общих к-меров, шаг выборки к-меров и минимальный размер кластера.

Каждый эксперимент включал в себя следующие этапы:

1) фильтрация повторов программой с соответствующим набором параметров и подготовка данных к сборке генома;

2) сборка подготовленных данных программой SPAdes;

3) анализ кодирующих данных методом выравнивания ридов РНК на полученную сборку;

4) выравнивание отфильтрованных ридов на базу данных повторов модельного растения Arabidopsis Thaliana.

Результаты экспериментов приведены в табл. 2.

Таблица 2 / Table 2

Результаты экспериментов

Results of the experiments

Параметр / Parameter	Эксперимент / Experiment, №				
	1	2	3	4	5
Размер исходных данных. МБ	2094				
Размер фильтрованных данных, МБ	1483	1460	1495	1571	1507
Часть ридов-повторов, %	26,6	28,2	26,5	23,2	26.1
Время фильтрации, мин:с	7:19	4:02	3:38	4:01	3:41
Расход памяти, ГБ	11,9	14,3	13,0	13,0	13,0
Количество используемых ядер	20				
Длина к-мера, п.н.о.	8	10	9	9	9
Количество общих к-меров	4	3	4	5	4
Шаг выборки к-меров	3	6	4	3	3
Минимальный размер кластера	30	20	25	40	30
Полная длина, млн п.н.о.	110,7	109,6	111,1	112,9	111,1
Количество контигов, тысяча	45,9	42,7	43,6	45,1	44.0
Максимальная длина контига, тысяча п.н.о.	65,6	55,7	58,8	75,9	83,3
Выравнивание ридов РНК на полученную сборку генома, %	87,66	87,55	87,61	87,65	87,67
N50, п.н.о.	7087	7575	8085	8856	7937

Лучший результат достигнут в эксперименте № 5 (табл. 2), показана наименьшая потеря кодирующей информации (выравнивание ридов РНК на полученную сборку генома – 87,67 %) при среднем значении процента отфильтрованных данных (1507 Мб) и практически минимальных временных затратах на фильтрацию ридов (3 мин 41 с). Сравнение результатов сборок генома (табл. 3) без фильтрации повторов и с фильтрацией повторов с параметрами, как в эксперименте № 5, показало значительное сокращение времени сборки: с 4 час 21 мин до 1 часа 58 мин при времени фильтрации 3 мин 41 с.

Таблица 3 / Table 3

Сравнение данных до и после эксперимента № 5

Comparison of assembling result before and after experiment # 5

Параметр / Parameter	Данные / Data	
	До	После
Расход памяти ОЗУ, Гб	106,8	76,6
Время сборки, час:мин:с	4:21:55	1:58:52
Полная длина, млн п.н.о.	121,4	111,1
Количество контигов, тысяча	106,5	44,0
Максимальная длина контига, тысяча п.н.о.	83,6	83,3
Выравнивание ридов РНК на полученную сборку генома, %	87,94	87,67
N50, п.н.о.	8341	7937

Экспериментально выведены оптимальные параметры (табл. 4) для «мягкой» фильтрации – т. е. такой фильтрации, при которой часть ридов-повторов попрежнему остается в исходных данных, но при этом потеря уникальных ридов минимальна и сохраняется высокая производительность фильтрации.

Параллельная реализация алгоритма позволила ускорить в 17 раз этап кластеризации ридов и более чем в 1,3 раза этап построения СИРК.

Таблица 4 / Table 4

Оптимальные параметры, полученные экспериментальным путем
Optimal parameters by an experimental approach

Длина к-мера, п.н.о.	9
Количество общих к-меров	4
Шаг выборки к-меров	3
Минимальный размер кластера	Покрытие генома, умноженное на 3

Заключение

Разработанный алгоритм позволяет значительно сократить объем входных данных для de Novo сборки генома без потери кодирующей информации, и как результат – уменьшить время ассемблирования. Тесты на модельном растении Arabidopsis Thaliana показали изменение количества кодирующей информации в пределах 0,005 % по отношению к ее исходному объему при уменьшении исходных данных на 25 %. Реализована возможность осуществлять тонкую настройку очистки входных данных и задавать требуемые системные ресурсы. Получено авторское свидетельство о государственной регистрации программ для ЭВМ [7].

ЛИТЕРАТУРА

1. The whole de novo genome sequencing and assembly of Siberian larch (Larix sibirica Ledeb.) and Siberian pine (Pinus sibirica Du Tour.) / N.V. Oreshkova, Yu.A. Putintseva, D.A. Kuzmin, V.V. Sharov, V.V. Biryukov, S.V. Makolov, K.O. Deych, A.A. Ibe, E.A. Shilkina, K.V. Krutovsky // The 3rd International Conference "Plant Genetics, Genomics, Bioinformatics and Biotechnology" PlantGen 2015: Abstract book. – Novosibirsk, 2015. – P. 37.
2. **Compeau P.E.C., Pevzner P.A., Tesler G.** How to apply de Bruijn graphs to genome assembly [Electronic resource] // Jorunal of Nature Biotechnology. – 2011. – Vol. 29, N 11. – Available at: http://www.nature.com/nbt/journal/v29/n11/full/nbt.2023.html (accessed: 09.01.2017).
3. Геномный ассемблер SPAdes [Электронный ресурс]. – URL: http://bioinf.spbau.ru/ru/spades (дата обращения: 09.01.2017).
4. ABySS – Canada's Michael Smith Genome Sciences Centre [Electronic resource]. – Available at: http://www.bcgsc.ca/platform/bioinfo/software/abyss (accessed: 09.01.2017).
5. What is FASTA format? [Electronic resource]. – Available at: http://zhanglab.ccmb.med.umich.edu/FASTA/ (accessed: 09.01.2017).
6. **Maumus F., Quesneville H.** Deep investigation of *Arabidopsis thaliana* junk DNA reveals a continuum between repetitive elements and genomic dark matter [Electronic resource] // PloS One. – 2014. – Vol. 9 (4). – Available at: http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0094101 (accessed: 09.01.2017).
7. Программный комплекс фильтрации повторяющихся последовательностей (рипитов) в ридах NGS Illumina: а.с. № 2015619173 Российская Федерация / А.Н. Цыбин, Д.А. Кузьмин, С.И. Феранчук, Ю.А. Путинцева; заявитель и правообладатель ФГАОУ ВПО «Сибирский федеральный университет» (СФУ). – Заявл. 01.07.2015; опубл. 26.08.2015.

PARALLEL REPEATS FILTRATION ALGORITHM OF NGS ILLUMINA DATA

Cybin A.N.[1], Sharov V.V.[1], Putinceva Ju.A.[2], Feranchuk S.I.[1,3], Kuz'min D.A.[1]
[1]*Siberian Federal University, Krasnoyarsk, Russia*
[2]*Institute of Forest of the Siberian Division of the Russian Academy of Sciences, Novosibirsk, Russia*
[3]*Irkutsk State Technical University, Irkutsk, Russia*

The approach on a preprocessing of NGS reads is considered which allows to reduce significantly a volume of input data of genome assembly for large genomes. The idea of the approach is

a filtering of reads which are parts of repeated elements in a genome. These parts of the genome are not used in the analysis of proteins encoded by the genome. The parallel probabilistic filtering algorithm is implemented, which allows to reduce significantly a time of de novo assembly with a minimal loss of coding information. The implementation of the algorithm is adjusted for a maximal performance. The approach was tested on the model plant Arabidopsis Thaliana with genome size 157 mln b.p. SPAdes genome assembler was used for assembly tests. The transcriptome mapping was used for the verification of the result. The size of an input NGS data for the assembly was reduced for more than 20 % after the preprocessing, the running time of the assembler was reduced more than twice and the loss of coding information was 0,005 %

Keywords: parallel algorithm, clustering, bioinformatics, repeats, filtration, sequence assembly, Illumina, SPAdes, Abyss.

DOI: 10.17212/1727-2769-2016-4-99-110

REFERENCES

1. Oreshkova N.V., Putintseva Yu.A., Kuzmin D.A., Sharov V.V., Biryukov V.V., Makolov S.V., Deych K.O., Ibe A.A., Shilkina E.A., Krutovsky K.V. The whole de novo genome sequencing and assembly of Siberian larch (Larix sibirica Ledeb.) and Siberian pine (Pinus sibirica Du Tour.). *The 3rd International Conference "Plant Genetics, Genomics, Bioinformatics and Biotechnology" PlantGen 2015*: Abstract book. Novosibirsk, 2015, p. 37.
2. Compeau P.E.C., Pevzner P.A., Tesler G. How to apply de Bruijn graphs to genome assembly. *Jorunal of Nature Biotechnology,* 2011, vol. 29, no. 11. Available at: http://www.nature.com/nbt/journal/v29/n11/full/nbt.2023.html (accessed 09.01.2017)
3. *Genomnyi assembler SPAdes* [Genomic assembler SPAdes]. Available at: http://bioinf.spbau.ru/ru/spades (accessed 09.01.2017)
4. *ABySS – Canada's Michael Smith Genome Sciences Centre*. Available at: http://www.bcgsc.ca/platform/bioinfo/software/abyss (accessed 09.01.2017)
5. *What is FASTA format?* Available at: http://zhanglab.ccmb.med.umich.edu/FASTA/ (accessed 09.01.2017)
6. Maumus F., Quesneville H. Deep investigation of Arabidopsis thaliana junk DNA reveals a continuum between repetitive elements and genomic dark matter. *PloS One*, 2014, vol. 9 (4). Available at: http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0094101 (accessed 09.01.2017)
7. Cybin A.N., Kuz'min D.A., Feranchuk S.I., Putintseva Yu.A. *Programmnyi kompleks fil'tratsii povtoryayushchikhsya posledovatel'nostei (ripitov) v ridakh NGS Illumina* [Software package for repeating sequences (repeats) filtration in NGS Illumina reads]. Inventor's Certificate RF, no. 2015619173, 2015.

СВЕДЕНИЯ ОБ АВТОРАХ

Цыбин Александр Николаевич – родился в 1994 году, магистрант, кафедра высокопроизводительных вычислений, СФУ. Область научных интересов: высокопроизводительные вычисления в задачах биоинформатики.

Cybin Alexander Nikolaevich (b. 1994) – master, high-performance computing department, Siberian Federal University. His research interests are currently focused on HPC in bioinformatics.

Шаров Вадим Витальевич – родился в 1992 году, научный сотрудник, НОЦ геномных исследований, СФУ. Область научных интересов: биоинформатика. Опубликовано 6 научных работ.

Sharov Vadim Vitalevich (b. 1992) – Researcher, Genome Research and Education Center, Siberian Federal University. His research interests are currently focused on bioinformatics. He is author of 6 scientific papers.

Путинцева Юлия Андреевна – родилась в 1985 году, научный сотрудник, Лаборатория лесоведения и почвоведения, Институт леса им. В.Н. Сукачева СО РАН. Область научных интересов: геномика, биоинформатика, системная биология. Опубликованы 32 научные работы.

Putintseva Yuliya Andreyevna (b. 1985) – Researcher, Laboratory of Forest and Soil Sciences, Institute of Forest of the Siberian Division of the Russian Academy of Sciences. Her research interests are currently focused on genomics, bioinformatics, system biology. She is author of 32 scientific papers.

Феранчук Сергей Ильич – родился в 1968 году, канд. физ.-мат. наук, научный сотрудник, научно-образовательный центр геномных исследований, Сибирский Федеральный университет, доцент; кафедра информатики, Иркутский национальный исследовательский технический университет. Опубликовано 25 научных работ. Область интересов: геномика, биоинформатика.

Feranchuk Sergey Il'ich (b. 1968) – Candidate of Sciences (Phys.&Math.), Researcher, Genome Research and Education Center, Siberian Federal University, Associated Professor, Department of informatics, Irkutsk State Technical University. His research interests are currently focused on genomics, bioinformatics. He is author of 25 scientific papers.

Кузьмин Дмитрий Александрович – родился в 1968 году, канд. техн. наук, заведующий кафедрой «Высокопроизводительные вычисления» в Сибирском Федеральном университете. Область научных интересов: суперкомпьютеры, высокопроизводительные вычисления. Опубликовано 30 научных работ. (Адрес: 660113, Российская Федерация, Красноярск, ул. Карбышева, 18. E-mail: dkuzmin@sfu-kras.ru).

Kuzmin Dmitry Alexandrovich (b. 1968) – Candidate of Sciences (Eng.), head of high-performance computing department. His research interests are currently focused on high performance computing. He is author of 30 scientific papers. (Address: Russian Federation, Krasnoyarsk, Karbyshev st., 18, E-mail: dkuzmin@sfu-kras.ru).

Статья поступила 04 июля 2016 г.
Received July 04, 2016

To Reference:

Cybin A.N., Sharov V.V., Putinceva Yu.A., Feranchuk S.I., Kuz'min D.A. Parallel'nyi algoritm fil'tratsii povtorov v dannykh NGS ILLUMINA [Parallel repeats filtration algorithm of NGS ILLUMINA data]. *Doklady Akademii nauk vysshei shkoly Rossiiskoi Federatsii – Proceedings of the Russian higher school Academy of sciences*, 2016, no. 4 (33), pp. 99–110. doi: 10.17212/1727-2769-2016-4-99-110