Компьютерные инструменты в образовании, 2016
№ 3: 5–15
УДК: 004.021
http://ipo.spb.ru/journal

информатика

АНАЛИЗ ГЕНОМНЫХ И МЕТАГЕНОМНЫХ ДАННЫХ В ОБРАЗОВАТЕЛЬНЫХ ЦЕЛЯХ

Казаков С.В.[1], Шалыто А.А.[1]

[1]Университет ИТМО, Санкт-Петербург, Россия

Аннотация

В работе рассматриваются две задачи анализа данных геномного и метагеномного секвенирования — задача *de novo* сборки генома (сборка неизвестного генома) и задача сравнительного анализа метагеномов, которая возникает при анализе геномов микроорганизмов из почв, океанов, кишечника человека и т. д. Несмотря на то, что эти задачи в основном возникают у исследователей, работающих в области биологии, *их использование в образовательных целях* — необходимый шаг при обучении молодых медиков, биологов и биоинформатиков, а также для повышения квалификации специалистов из этих областей. В настоящей статье приводится обзор методов сборки генома и сравнительного анализа метагеномов, исследуется вопрос применимости существующих средств для обучающихся и предлагаются новые подходы к решению данных задач. Такие подходы использовались авторами при обучении студентов в Санкт-Петербургском политехническом университете Петра Великого. В работе также приводятся результаты экспериментов по сравнению предложенных подходов с известными.

Ключевые слова: *биоинформатика, ДНК, геном, метагеном, секвенирование ДНК, сборка генома de novo, сравнительный анализ метагеномов, персональный компьютер.*

Цитирование: Казаков С.В., Шалыто А.А. Анализ геномных и метагеномных данных в образовательных целях // Компьютерные инструменты в образовании, 2016. № 3. С. 5–15 .

1. ВВЕДЕНИЕ

Исследования в биологических областях с каждым днем все больше влияют на наши знания о фундаментальных законах работы клеток организмов и взаимодействии между ними.

Для полноценного понимания «законов жизни» создаются модели явлений и процессов, происходящих как в клетке, так и на других уровнях. К современным методам получения информации о клетке относится *секвенирование ДНК* — определение последовательности нуклеотидов в молекуле ДНК (дезоксирибонуклеиновая кислота). Эта молекула обеспечивает хранение и передачу генетической информации.

Один из первых методов секвенирования был предложен Ф. Сэнгером в 1977 г., впоследствии названный методом Сэнгера, или методом обрыва цепи.

С середины первого десятилетия XXI века широкое распространение получили так называемые *технологии секвенирования нового поколения* (*next generation sequencing*, *NGS*) [1]. В отличие от предшествующих методов, эти технологии позволяют получать десятки и сотни гигабайт исходных данных, вместе с тем уменьшая длину единичного прочтения с 1000 нуклеотидов до 30–300 и вводя так называемые «парные чтения» вместо ранее использованных непарных.

По мере развития технологий секвенирования создавались и программы для их анализа. Такие программы в основном были предназначены для узкого круга лиц — ученых или специалистов, которые понимают, как они должны работать, и могут в них разобраться и запустить. При этом подавляющее большинство программ для биоинформатического анализа создаются под операционную систему *Linux* и ориентированы для работы *на серверах* с большим объемом оперативной памяти. Это связано с тем, что для биоинформатического анализа регулярно требуется обработка больших данных (десятки и сотни гигабайт). Однако использование таких программ для обучения — сложная задача, так как установка, настройка и запуск их на персональных компьютерах обучающихся плохо осуществимы.

Сборка геномной последовательности (или коротко — *сборка генома*) — процесс получения больших фрагментов генома из небольших чтений, полученных при секвенировании ДНК. Задача *de novo сборки генома* — это задача сборки неизвестного генома.

Задача сборки геномных последовательностей *de novo* является, в определенном смысле, центральной среди всех задач биоинформатики [2]. Это объясняется тем, что без ее решения нельзя приступить к детальному изучению генома и его анализу с применением других алгоритмов биоинформатики.

Для решения задач по сборке генома к настоящему времени были разработаны десятки алгоритмов сборки [3–9]. Реализовав предложенные алгоритмы в программы (*сборщики*), многие из них сейчас широко распространены и популярны (например *ABySS*, *Velvet* [9], *SPAdes* [8], *Newbler* (Roche, Швейцария) и т. д.). При этом такие решения требуют большого объема оперативной памяти для работы. Пытаться собрать геном на персональном компьютере с помощью этих сборщиков — практически невыполнимая задача [7, 10].

Единственным известным сборщиком, ориентированным на работу на персональных компьютерах (в том числе и под *Windows*), является *CLC Genomics Workbench* [11], который не распространяется свободно, является платным и весьма дорогим.

Таким образом, для использования программ по *de novo* сборке генома в образовательных целях необходимо выполнение следующих критериев:

- Возможность работы сборщика на персональных компьютерах обучающихся (при небольших объемах оперативной памяти и под управлением распространенных операционных систем *Windows*, *Linux*, *OS X*/*Mac OS*).
- Возможность простого и понятного запуска сборщика даже непрофессионалами (наличие графического интерфейса пользователя).

В работе производится сравнение существующих и предлагаемого подходов по указанным критериям.

2. СБОРКА ГЕНОМА *DE NOVO*

Сборка генома — процесс восстановления исходного генома из небольших чтений, полученных при секвенировании ДНК. С помощью таких чтений исходный геном обычно покрывают несколько десятков раз.

Обычно решением этой задачи является набор контигов. *Контигом* называется непрерывная последовательность нуклеотидов, которую не удается расширить.

Контиги могут быть объединены в скэффолды. *Скэффолдом* называется последовательность контигов с оценкой расстояний между ними.

Существует и более законченный результат сборки — *референсная последовательность* (*референс*). Под *референсом* понимается собранный геном высокого качества (один контиг или один скэффолд).

Процесс сборки генома обычно разбит на несколько этапов:

– исправление ошибок в исходных данных;
– восстановление небольших фрагментов генома по парным чтениям, такие фрагменты называются *квазиконтигами*;
– сборка контигов (расширенных фрагментов) из квазиконтигов;
– построение скэффолдов.

2.1. Существующие решения

Современные сборщики генома работают с графовым представлением задачи — это и удобно в представлении, и практично для написания алгоритмов. Существует две основные структуры для графового представления — *граф перекрытий* (*overlap graph*) и *граф де Брёйна* (*de Bruijn graph*).

Графом перекрытий называется взвешенный ориентированный граф, каждой вершине которого сопоставлена строка s_i (исходные чтения), а ребро между двумя вершинами проводится, если соответствующие строки перекрываются (строка a перекрывается со строкой b, если непустой суффикс a совпадает с префиксом b). Вес ребра в таком случае — длина перекрытия. Пример графа перекрытий показан на рис. 1 (a).

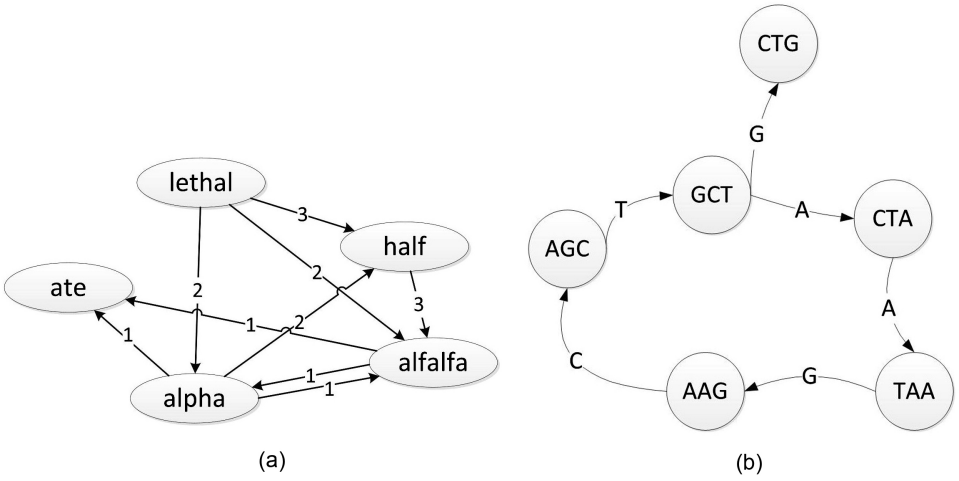

Рис. 1. Примеры графового представления задачи для (a) графа перекрытий, (b) графа де Брёйна

Графом де Брёйна степени k над алфавитом Σ называется ориентированный граф, в котором вершинами являются строки фиксированной длины k (k-меры) из Σ^k, а ребрами являются строки e длины $k+1$ из Σ^{k+1}, причем ребро e соединяет вершины $e[1..k]$ и $e[2..k+1]$. Иными словами, ребро между двумя вершинами проводится, если из первого k-мера можно получить второй путем добавления одного символа в конец первого

k-мера и убирания одного символа из начала. Пример графа де Брёйна для $k = 3$ показан на рис. 1 (b).

Несмотря на разницу в представлениях данных в графах перекрытий и де Брёйна, все основные шаги по сборке могут с небольшими изменениями работать на каждом из них. Разница состоит в том, насколько удобно и компактно хранятся исходные данные в этих структурах. Например, граф де Брёйна рекомендуется использовать при большом числе исходных чтений и их небольшой длине — это обеспечивает уменьшение используемой памяти. Граф перекрытий рекомендуется использовать при «длинных» чтениях — это позволяет более полно использовать всю имеющуюся информацию.

В таблице 1 представлена информация о широко известных сборщиках генома. Информация была взята с официальных сайтов сборщиков или из руководств по их использованию. В таблицу сравнения также добавлено предлагаемое решение — *ITMO Genome Assembler* [3–5].

Таблица 1. Сравнение существующих сборщиков

Название	Годы разработки	Основан на	Поддержив. операцион. системы	Дополнительная информация
CLC Genomics Workbench [11]	2008-2016	Граф де Брёйна	*Linux*, *OS X*/*Mac OS*, *Windows*	Возможно проведение сборки на персональном компьютере. Возможность запуска через графический интерфейс. **Коммерческая лицензия**.
ITMO Genome Assembler [3–5] (предлагаемое решение)	2010-2016	Граф де Брёйна, граф перекрытий	*Linux*, *OS X*/*Mac OS*, *Windows*	Возможно проведение сборки на персональном компьютере. Возможность запуска через графический интерфейс. Распространяется свободно.
MaSuRCA [6]	2012-2015	Граф де Брёйна, граф перекрытий	*Linux*	Распространяется свободно.
Minia [7]	2012-2016	Сжатый граф де Брёйна	*Linux*, *OS X*/*Mac OS*	Возможно проведение сборки на персональном компьютере. Распространяется свободно.
Newbler (Roche, Швейцария)	2009-2013	Граф перекрытий	*Linux*	Возможность запуска через графический интерфейс. Распространяется свободно.
SPAdes [8]	2012-2016	Граф де Брёйна	*Linux*, *OS X*/*Mac OS*	Распространяется свободно.
Velvet [9]	2007-2013	Граф де Брёйна	*Linux*, *OS X*/*Mac OS*	Распространяется свободно.

2.2. Предлагаемое решение

В лаборатории «Компьютерные технологии» Университета ИТМО был разработан сборщик генома *ITMO Genome Assembler* [3–5], состоящий из набора алгоритмов на языке программирования *Java* для выполнения сборки *de novo* на персональных компьютерах и под управлением широко распространенных операционных систем (*Windows*, *Linux*, *OS X*/*Mac OS*).

Сборщик основан на совместном применении графов де Брёйна и графов перекрытий, что позволяет использовать преимущества обеих структур данных.

Схема предлагаемого решения показана на рис. 2.

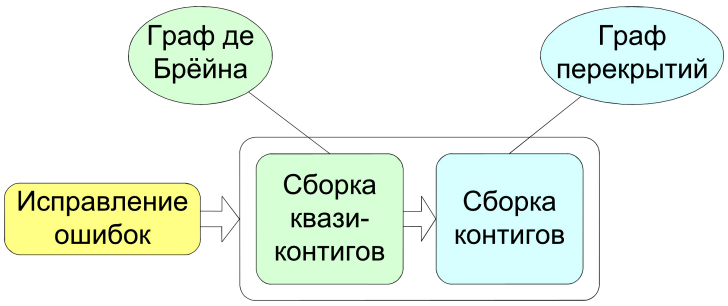

Рис. 2. Схема предлагаемого решения

Подробное описание используемых подходов приведено в работах [3–5]. Ниже приводится их краткое описание.

Алгоритм исправления ошибок основан на частотном анализе содержащихся в чтениях k-меров (подстрок длины k) и не использует граф де Брёйна. Для эффективного исправления ошибок необходимо, чтобы каждая позиция генома была прочитана несколько раз. Тогда одинаковые k-меры без ошибок будут встречаться несколько раз в исходных чтениях, тогда как k-меры с ошибками будут встречаться достаточно редко. Данное наблюдение основано на том, что ошибки при чтении возникают на случайных позициях, и, следовательно, вероятность того, что найдутся два k-мера с одинаковой ошибкой, достаточно мала. Метод пытается исправить все ошибочные k-меры (редко встречаемые) на безошибочные путем изменения нуклеотидов на отдельных позициях.

Сборка квазиконтигов использует *граф де Брёйна*, который строится из k-меров, встречающихся в чтениях достаточно большое число раз. Метод сборки квазиконтигов пытается восстановить фрагмент, из которого были получены парные чтения. Более точно: происходит поиск такого пути в графе де Брёйна, который мог бы соответствовать фрагменту генома, из которого, вероятно, были получены парные чтения. Из всех таких путей нас интересуют только укладывающиеся в априорные границы длин фрагментов. Поэтому слишком короткие и слишком длинные пути можно далее не рассматривать. Если нашелся единственный подходящий путь, то можно с очень большой уверенностью сказать, что он соответствует реальной подстроке геномной последовательности, поэтому этот фрагмент считается восстановленным, а найденный путь выводится в качестве квазиконтига.

Сборка контигов из квазиконтигов основана на подходе *overlap-layout-consensus* (*OLC*) с применением *графа перекрытий*. Используемый подход состоит в последовательном выполнении нескольких этапов — поиск перекрытий между всеми квазиконтигами (*overlap*), построение графа перекрытий по ним, упрощение графа перекрытий, поиск путей в графе перекрытий (*layout*) и вывод консенсуса для путей (*consensus*). Для каждого пути вычисляется консенсус чтений, участвующих в пути (для каждой позиции выбирается нуклеотид, который всех чаще встречается), получившуюся последовательность записывают как контиг.

Данный подход является известным. Он применяется при сборке генома на основе графа перекрытий. Отличительной особенностью сборки контигов в предлагаемом решении является то, что каждый из указанных выше этапов разбивается на несколько

подэтапов в случае, если доступной оперативной памяти не хватает для выполнения всего этапа. При этом на выполнение каждого из подэтапов требуется меньше памяти, чем для всего этапа в целом. После выполнения подэтапа память освобождается.

Новизной предлагаемого сборщика является то, что он использует разные графы на разных этапах сборки. На этапе сборки квазиконтигов из парных чтений лучше использовать граф де Брёйна, так как данных может быть много, а длина чтений при этом может быть небольшой. На этапе сборки контигов лучше использовать граф перекрытий, так как при этой сборке используются достаточно длинные квазиконтиги (в среднем по 500 нуклеотидов). Схожий подход используется и в сборщике *MaSuRCA*, однако авторами он был впервые предложен в 2011 г. на конференции *de novo Genome Assembly Assessment Project workshop* (*dnGASP*), в то время как сборщик *MaSuRCA* начал разрабатываться только в 2012 г., а первая статья с его описанием вышла в 2013 г. [6].

Для хранения графа де Брёйна используется хеш-таблица с открытой адресацией. При этом хранится только сам граф (вершины графа) без сохранения дополнительной информации в вершинах и на ребрах. Это позволяет минимизировать используемую память на первом шаге сборки и сильно сократить используемые ресурсы.

2.3. Экспериментальные исследования

Для экспериментальной проверки работы сборщиков были использованы два набора данных: **малый по размеру геном** — данные секвенирования бактерии *Escherichia coli* (кишечная палочка) и **средний по размеру геном** — 14-ой хромосомы человека.

Информация об используемых данных приведена в таблице 2.

Таблица 2. Используемые наборы данных

№	Название организма	Размер генома	Платформа секвенирования	Информация об исходных данных
1.	Escherichia coli (кишечная палочка)	4,6 млн. нукл.	*Illumina Genome Analyzer*	Парные чтения, размер фрагмента — 200 нукл., длина чтения — 36 нукл., покрытие исходного генома — 161,5x.
2.	Human chromosome 14 (14 хромосома генома человека)	107,35 млн. нукл.	*Illumina HiSeq* 2000	Парные чтения, размер фрагмента — 155 нукл., длина чтения — 101 нукл., покрытие исходного генома — 34,3x.

Для определения качества предлагаемого решения на приведенных данных было выполнено его сравнение с другими сборщиками (*SPAdes*, *Velvet*, *MaSuRCA*, *Minia*, *Newbler*). Сборщики *SPAdes*, *Velvet*, *MaSuRCA* и *Newbler* не смогли произвести сборку на *втором* наборе данных из-за нехватки оперативной памяти в 16 Гб. Сборщик *Newbler* не смог произвести сборку и на *первом* наборе данных, так как использованные чтения были слишком короткими для него. Сборщик *CLC Genomics Workbench* не использовался — он не распространяется свободно.

Сборщик *ITMO Genome Assembler* позволял собирать геном первого организма и при следующих ограничениях оперативной памяти: 2 Гб, 1 Гб, 0.5 Гб, а второго — при 4 Гб. Для

каждого их этих ограничений производился отдельный эксперимент. Остальные сборщики, в отличие от предлагаемого, *не имеют возможности задавать объем используемой памяти*.

Результаты экспериментов приведены в таблицах 3 и 4. *Параметр N50* — медианное значение длин собранных контигов (такая длина, что все контиги с большей длиной покрывают 50% итоговой сборки).

Таблица 3. Сравнение сборщиков на первом наборе данных (*E.coli*, геном 4,6 млн. нукл., суммарно 4518 известных генов)

Сборщик	Использ. память, Гб	Время работы, мин:сек	Число контигов	N50, тыс. нукл.	Процент собранного генома	Собрано генов
SPAdes	2.90	19:35	127	82.4	98.0	4378 + 36 part
Velvet	2.13	7:32	110	95.4	97.6	4360 + 39 part
MaSuRCA	3.75	11:08	166	54.3	97.9	4355 + 91 part
Minia	1.21	0:59	461	16.3	97.2	4163 + 172 part
ITMO Genome Assembler (2 Гб)	2.30	24:11	246	37.1	98.2	4350 + 116 part
ITMO Genome Assembler (1 Гб)	1.17	12:11	247	35.8	98.1	4347 + 120 part
ITMO Genome Assembler (0.5 Гб)	0.60	6:40	270	32.3	98.1	4344 + 119 part

Таблица 4. Сравнение сборщиков на втором наборе данных (14-я хромосома человека, геном 107,3 млн. нукл., суммарно 2244 известных генов).

Сборщик	Использ. память, Гб	Время работы, ч:мин:сек	Число контигов	N50, тыс. нукл.	Процент собранного генома	Собрано генов
Minia	4.11	0:26:20	37017	2.3	60.3	682 + 1164 part
ITMO Genome Assembler (4 Гб)	4.63	1:51:06	37012	2.9	71.9	886 + 1184 part

Тестирование сборщиков проводилась на компьютере с 16 ГБ оперативной памяти и 6-ядерным процессором *AMD PhenomTM II X6 1090T* под управлением *OS Linux 3.13.0 x86_64*. Эта операционная система использовалась для проведения экспериментов, так как часть сборщиков не может работать на других операционных системах. При этом реально используемый объем памяти контролировался средствами операционной системы *Linux*, и пик объема записывался в таблицу.

Из представленных таблиц можно сделать следующие выводы:

- При сборке бактериального генома качественнее всего собирает *SPAdes* (по числу найденных генов), однако он требует 2,9 Гб оперативной памяти.
- Ни один из известных сборщиков *SPAdes*, *Velvet*, *MaSuRCA*, *Newbler* не способен произвести сборку среднего по размеру генома при ограничении памяти в 16 Гб.

- Сборщик *Minia* работает хорошо и быстро, однако качество сборки (медиана длин контигов, покрытие исходного генома, число найденных генов) невысокое.
- Сборщик *ITMO Genome Assembler* работает достаточно хорошо и использует небольшой объем оперативной памяти. Сравнение со сборщиком *Minia* (использующим сжатое представление графа де Брёйна) показывает, что расход памяти на используемых наборах в целом одинаковый, однако качество сборки у *ITMO Genome Assembler* выше. Кроме того *ITMO Genome Assembler* является единственным сборщиком, работающим под ОС *Windows*.

3. СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТАГЕНОМОВ

Первоначально секвенирование ДНК применялось для определения последовательности ДНК отдельного организма. Со временем появилось *метагеномное секвенирование* — «чтение» ДНК всех организмов, находящихся в среде (без изолирования конкретных микроорганизмов) [12]. Совокупность таких последовательностей получила название *метагеном*. Это направление в биоинформатике также должно использоваться при обучении, что достигается проведением анализа небольших метагеномов на персональных компьютерах.

При анализе набора метагеномов важно оценивать степень сходства или различия разных метагеномов. Область знаний, изучающая данный вопрос, получила название — *сравнительная метагеномика*.

Сравнительную метагеномику также интересуют вопросы выделения конкретных признаков, которые отличают одно сообщество микроорганизмов от другого. Такими признаками могут быть, например, сами организмы или их ДНК, которые присутствуют в одном метагеноме и отсутствуют в другом. Также признаком может быть разная представленность некоторых микроорганизмов в разных сообществах.

Существует много подходов для сравнительного анализа метагеномов. Всех их можно разделить на несколько классов. Среди них — традиционные методы, основанные на *выравнивании метагеномных данных* на каталог известных геномов [13, 14]; методы, основанные на *совместной сборке* метагеномных данных [15]; абстрактные методы, базирующиеся на *выделении и анализе k-мерного спектра* набора метагеномов [16] и т. п. Однако все они имеют существенные недостатки, которые ограничивают область их применимости. Например, традиционные методы требуют репрезентативной базы геномов, однако многие виды бактерии до сих пор не изучены и не имеют референсной последовательности. Методы, основанные на совместной сборке данных секвенирования, требуют больших вычислительных ресурсов [17] и труднопримени мы в случае метагеномов со сложным бактериальным составом (таких, как микробиота кишечника человека). Абстрактные методы, основанные на анализе *k*-мерного спектра, не позволяют получать информативные признаки, которые будут отличать один метагеном от другого.

На основе описанного выше метода сборки генома *de novo* был предложен метод для сравнительного анализа метагеномов. Он в целом схож с подходами, основанными на совместной сборке метагеномов, однако, в отличие от них, предложенный метод не производит сборку целиком, а только выполняет «частичную» сборку. Благодаря этому удается избавиться от главных недостатков таких методов — требования больших вычислительных ресурсов. В отличие от других методов, предложенный подход позволяет получать информативные признаки и не требует базы референсных геномов.

Полное описание предложенного метода и экспериментальное сравнение с существующими решениями подробно описано в работе [17].

Следует отметить, что такое решение тоже написано на языке *Java* и является кроссплатформенным, поэтому разработанная программа может быть запущена на всех распространенных операционных системах — *Windows*, *Linux*, *OS X*/*Mac OS*. Из изложенного следует, что предложенное решение, как и предыдущее, может быть использовано в образовательном процессе.

4. ЗАКЛЮЧЕНИЕ

В работе были рассмотрены возможности использования решений задач *de novo* сборки генома и сравнительного анализа метагеномов для образования. Были описаны предложенные подходы, а также проведено экспериментальное исследование для сравнения возможностей подходов.

Разработанные методы по сборке генома *de novo* успешно применялись при выполнении лабораторных работ в рамках магистерской программы «Прикладная математика и информатика. Биоинформатика» на кафедре прикладной математики Санкт-Петербургского политехнического университета Петра Великого.

Список литературы

1. *Schuster S.C.* Next-generation sequencing transforms today's biology // Nature, 2007. № 200(8). P. 16–18.
2. *Miller J.R., Koren S., Sutton G.* Assembly algorithms for next-generation sequencing data // Genomics, 2010. № 95(6). P. 315–327.
3. *Сергушичев А.А., Александров А.В., Казаков С.В., Царев Ф.Н., Шалыто А.А.* Совместное применение графа де Брёйна, графа перекрытий и микросборки для de novo сборки генома // Известия Саратовского университета. Новая серия. Серия Математика. Механика. Информатика, 2013. № 13(2-2). С. 51–57.
4. *Александров А.В., Казаков С.В., Мельников С.В., Сергушичев А.А., Царев Ф.Н.* Метод сборки контигов геномных последовательностей на основе совместного применения графов де Брюина и графов перекрытий // Научно-технический вестник информационных технологий, механики и оптики, 2012. № 6 (82). С. 93–98.
5. *Alexandrov A., Kazakov S., Melnikov S., Sergushichev A., Shalyto A., Tsarev F.* Combining de Bruijn graph, overlaps graph and microassembly for de novo genome assembly // Proceedings of "Bioinformatics", 2012. P. 72.
6. *Zimin A.V., Marçais G., Puiu D., Roberts M., Salzberg S.L., Yorke J.A.* The MaSuRCA genome assembler // Bioinformatics, 2013. № 29(21). P. 2669–2677.
7. *Chikhi R., Rizk G.* Space-efficient and exact de Bruijn graph representation based on a Bloom filter // Algorithms for Molecular Biology, 2013. 8(1). P. 22.
8. *Bankevich A., Nurk S., Antipov D., Gurevich A.A., Dvorkin M., Kulikov A.S., Lesin V.M., Nikolenko S.I., Pham S., Prjibelski A.D., Pyshkin A.V.* SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing // Journal of Computational Biology, 2012. № 19(5). P. 455–477.
9. *Zerbino D.R., Birney E.* Velvet: algorithms for de novo short read assembly using de Bruijn graphs // Genome research, 2008. № 18(5). P. 821–829.
10. *Kleftogiannis D., Kalnis P., Bajic V.B.* Comparing memory-efficient genome assemblers on stand-alone and cloud infrastructures // PloS one, 2013. № 8(9). e75505.
11. CLC Genomics Workbench — QIAGEN Bioinformatics. https://www.qiagenbioinformatics.com/products/clc-genomics-workbench (дата обращения: 05.06.2016).

12. *Handelsman J., Rondon M.R., Brady S.F., Clardy J., Goodman R.M.* Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products // Chemistry & biology, 1998. № 5(10). P. R245–R249.
13. *Wood D.E., Salzberg S.L.* Kraken: ultrafast metagenomic sequence classification using exact alignments // Genome biology, 2014. № 15(3). R46.
14. *Truong D.T., Franzosa E.A., Tickle T.L., Scholz M., Weingart G., Pasolli E., Tett A., Huttenhower C., Segata N.* MetaPhlAn2 for enhanced metagenomic taxonomic profiling // Nature methods, 2015. № 12(10). P. 902–903.
15. *Dutilh B.E., Schmieder R., Nulton J., Felts B., Salamon P., Edwards R.A., Mokili J.L.* Reference-independent comparative metagenomics using cross-assembly: crAss // Bioinformatics, 2012. № 28(24). P. 3225–3231.
16. *Wu Y.W., Ye Y.* A novel abundance-based algorithm for binning metagenomic sequences using l-tuples // Journal of Computational Biology, 2011. 18(3), P. 523–534.
17. *Ulyantsev V.I., Kazakov S.V., Dubinkina V.B., Tyakht A.V., Alexeev D.G.* MetaFast: fast reference-free graph-based comparison of shotgun metagenomic data // Bioinformatics, 2016. btw312.

Поступила в редакцию 16.05.2016, окончательный вариант — 08.06.2016.

Computer tools in education, 2016
№ 3: 5–15
http://ipo.spb.ru/journal

GENOME AND METAGENOME DATA ANALYSIS FOR EDUCATION

Kazakov S.V.[1], Shalyto A.A.[1]

[1]ITMO University, Saint-Petersburg, Russia

Abstract

In this paper we address two problems of analyzing genome and metagenome sequencing data — *de novo* genome assembly problem (assembly of an unknown genome) and problem of comparative metagenome analysis which arises in the analysis of microorganisms in soil, sea, human gut, etc. Despite these problems are of interest to scientists working in the biology area, *using them for education* is essential for teaching medical students, biologists, bioinformaticians and also in the process of further training of specialists in this areas. In this paper we present a survey of methods for *de novo* genome assembly and comparative metagenome analysis, examine the possibility of using such approaches in educational processes and propose novel approaches for solving these problems. Proposed solutions have already been used for educating students in the Peter the Great St.Petersburg Polytechnic University. In this paper we also present the results of experiments of comparing proposed methods against known ones.

Keywords: *bioinformatics, DNA, genome, metagenome, DNA sequencing, de novo genome assembly, comparative metagenome analysis, personal computer.*

Citation: Kazakov S.V. & Shalyto A.A. 2016, “Analiz genomnykh i metagenomnykh dannykh v obrazovatel’nykh tselyakh” [“Genome and Metagenome Data Analysis for Education”], *Computer tools in education*, no 3, pp. 5–15.

Received 16.05.2016, the final version — 08.06.2016.

Sergey V. Kazakov, postgraduate student, Computer Technologies Department, ITMO University. 197101 Russia, Saint-Petersburg, Kronverksky Pr., 49. Computer Technologies Department. svkazakov.me@gmail.com
Anatoly A. Shalyto, Doctor of Science, Professor, Head of Programming Technologies Department, ITMO University, shalyto@mail.ifmo.ru

**Казаков Сергей Владимирович,
аспирант кафедры «Компьютерные технологии» Университета ИТМО,
197101 Санкт- Петербург, Кронверкский пр., 49, кафедра «Компьютерные технологии»,
svkazakov.me@gmail.com**

**Шалыто Анатолий Абрамович,
доктор технических наук, профессор,
заведующий кафедрой «Технологии Программирования» Университета ИТМО,
shalyto@mail.ifmo.ru**