Московский государственный университет им. М.В. Ломоносова
Факультет вычислительной математики и кибернетики

На правах рукописи

Sign

Романенков Кирилл Владимирович

Методы и алгоритмы автоматизированной сборки генома на вычислительном кластере

Специальность 05.13.11 —
«математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей»

Автореферат
диссертации на соискание учёной степени
кандидата физико-математических наук

Москва — 2018

Работа выполнена на кафедре суперкомпьютеров и квантовой информатики факультета вычислительной математики и кибернетики Федерального государственного бюджетного образовательного учреждения высшего образования «Московский государственный университет имени М.В. Ломоносова».

Научные руководители: чл.-кор. РАН, д.ф.-м.н.
Королев Лев Николаевич
чл.-кор. РАН, д.ф.-м.н.
Якобовский Михаил Владимирович

Официальные оппоненты: **Фамилия Имя Отчество,**
доктор физико-математических наук, профессор,
Не очень длинное название для места работы,
старший научный сотрудник

Фамилия Имя Отчество,
кандидат физико-математических наук,
Основное место работы с длинным длинным длинным длинным названием,
старший научный сотрудник

Ведущая организация: Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования с длинным длинным длинным длинным названием

Защита состоится «____» _____ 2018 года в _____ на заседании диссертационного совета Д 002.024.01 при Федеральном государственном учреждении «Федеральный исследовательский центр Институт прикладной математики им. М.В. Келдыша Российской академии наук» по адресу: 125047, Москва, Миусская пл., 4.

С диссертацией можно ознакомиться в библиотеке ИПМ им. М.В. Келдыша РАН, http://keldysh.ru.

Отзывы на автореферат в двух экземплярах, заверенные печатью учреждения, просьба направлять по адресу: 125047, Москва, Миусская пл., 4, ученому секретарю диссертационного совета Д 002.024.01.

Автореферат разослан «____» _____ 2018 года. Телефон для справок: 7 (499) 250-78-66.

Ученый секретарь
диссертационного совета
Д 002.024.01,
д.ф.-м.н., доцент Sign Бондарев Александр Евгеньевич

Общая характеристика работы

Актуальность темы. Расшифровка ДНК позволяет решать многие проблемы человеческого организма. Например, помогает проводить профилактику врожденных болезней или заболеваний, к которым человек был предрасположен изначально. Наличие подобной информации помогает гораздо быстрее ставить диагноз, выбирать методы лечения, а также прогнозировать исход болезни с бóльшей точностью.

Технологии расшифровки последовательностей ДНК значительно изменились с тех пор, как в 1977 году был прочитан первый геном (вируса–бактериофага $\phi\psi$174). В настоящее время они успешно применяются в:

- персонифицированной медицине
- производстве лекарственных средств
- генной инженерии
- теории эволюции
- и многих других научных областях.

Восстановление последовательностей хромосом изучаемого организма по-прежнему остается весьма сложной задачей. Несмотря на успехи в расшифровке геномов человека, мыши и других организмов, большая часть биоты еще даже не секвенирована. Кроме того, в случае больших геномов полностью восстановить последовательности хромосом не удается или удается с большим трудом. Так, последняя сборка *hg38* генома человека состоит из 735 «скэффолдов» — непрерывных последовательностей, — вместо 24-х в идеале. Число нерасшифрованных нуклеотидов оценивается в 160 млн. (расшифрованных — 3 млрд 209 млн 286 тыс 105). Показательно также, что среди геномов 1542 видов эукариот, заявленных в БД NCBI в разделе секвенированных геномов, лишь 14 отнесены к группе полностью секвенированных геномов.

Ключевой этап сборки генома *de novo* (то есть сборки, при которой отсутствует ранее восстановленный геном особи того же или родственного вида, что и исследуемая) — это склеивание прочитанных фрагментов генома («ридов») на основе их перекрывающихся участков. Другими словами, составление из перекрывающихся последовательностей, считанных из случайно выбранных мест генома, набора последовательностей большей длины («контигов»), которые соответствуют непрерывным фрагментам секвенируемой ДНК.

Было разработано большое количество программных продуктов для решения задачи сборки генома, среди которых можно выделить такие, как: Atlas, Arachne, Celera, PCAP, Phrap и Phusion. Все эти программы основаны на методе перекрытия («overlap-layout-consensus»). Этот способ применялся во множестве проектов сборки геномов и был очень распространен до появления методов секвенирования нового поколения (Next Generation Sequencing). Одна из особенностей метода перекрытия заключается в необходимости попарного сравнения прочитанных фрагментов генома.

Методы секвенирования нового поколения начали активно применяться для исследования геномов в середине 2000-х годов и позволили снизить стоимость секвенирования на несколько порядков. На рис. 1 показана тенденция к снижению стоимости секвенирования 1 мегабазы (1 млн) нуклеотидов. К

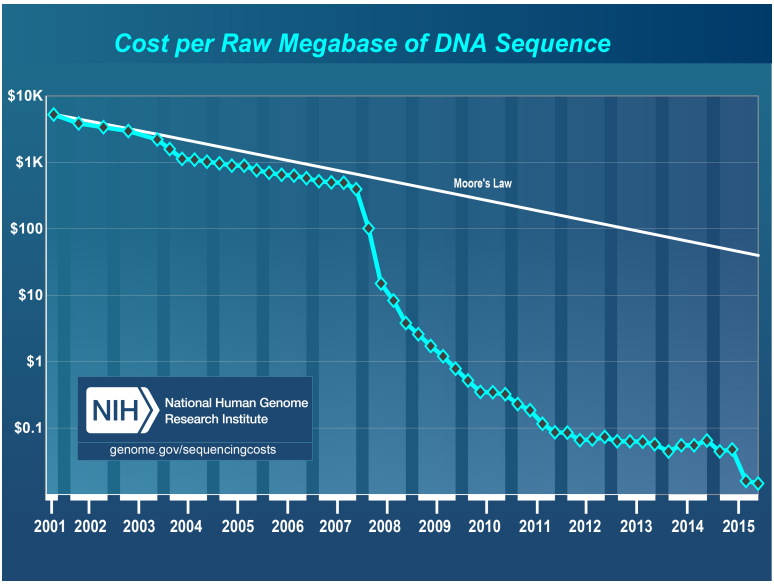


Рис. 1 — Стоимость секвенирования 1 млн пар нуклеотидных оснований. Логарифмическая шкала.

методам секвенирования нового поколения относят пиросеквенирование (454 Sequencing), секвенирование посредством синтеза (Illumina), секвенирование посредством лигирования (SOLiD) и секвенирование посредством полупроводникового чипа (Ion Torrent). По сравнению с традиционным методом Сэнгера эти технологии позволяют добиться значительно большей производительности.

Тем не менее, длина фрагментов генома, прочитанных с помощью методов секвенирования нового поколения, значительно меньше длины ридов, получаемых методом Сэнгера. Для пиросеквенирования она составляет около 700 нуклеотидных оснований, 100–300 оснований для Illumina, 60 оснований для секвенирования посредством лигирования и 200 оснований для секвенирования посредством полупроводникового чипа. Небольшой размер этих ридов затрудняет использование метода overlap-layout-consensus для сборки исследуемого генома. Из-за своей короткой длины риды должны быть многократно прочитаны с большей глубиной покрытия, нежели у более ранних проектов по секвенированию геномов. Огромное количество получаемых ридов делает вышеупомянутый подход к сборке, требующий попарного сравнения всех чтений, вычислительно крайне неэффективным. В то время как длинные риды могут формировать

длинные участки перекрытий, которые позволяют отличать ситуации реальных перекрытий от ситуаций, когда перекрытия возникают из-за наличия на концах фрагментов повторяющихся участков, короткие риды, содержащие повторы, гораздо реже позволяют установить истинную причину наличия перекрытия между фрагментами. Эти проблемы стимулировали разработку и исследование алгоритмов, предназначенных для сборки генома из коротких ридов.

Компания Pacific Biosciences (PacBio), которая также является крупным игроком на рынке секвенирования, придерживается другой идеологии. Ее технология Single Molecule Real-time (SMRT) позволяет получать риды значительно большей длины, чем при использовании традиционного метода Сэнгера, в среднем около 10–15 тыс пар нуклеотидных оснований, которые, однако, характеризуются большей частотой появления ошибок секвенирования.

Похожая концепция используется компанией Oxford Nanopore Technologies в своей технологии MinION. С помощью специального портативного устройства размером с USB-накопитель, можно проводить секвенирование ДНК в реальном времени, получая риды, имеющие в среднем длину около 2–5 тыс пар нуклеотидных оснований. Последовательности, получаемые с помощью этой технологии, содержат ошибки секвенирования с большей частотой, чем при использовании метода SMRT. Обе вышеперечисленные компании предлагают использовать специализированное программное обеспечение для сборки геномов, прочитанных с помощью их секвенаторов. Необходимо отметить, что зачастую в проектах расшифровки геномов используются риды, полученные с применением различных технологий: например, короткие чтения Illumina и длинные риды, полученные с помощью технологии SMRT.

Вне зависимости от используемой технологии получения ридов для работы геномных сборщиков – специальных программ для восстановления секвенированного генома – необходимо задавать большой набор параметров. В качестве примера таких параметров можно привести длину прочитанных фрагментов генома, величину наложения ридов, при которой перекрытие не считается вызванным случайным совпадением нуклеотидов, минимальную длину получаемых контигов, глубину покрытия генома чтениями, величины, связанные с организацией внутренних структур сборщика и прочее. Пользователю, который слабо знаком с деталями реализации алгоритма, используемого в том или ином сборщике, достаточно тяжело выбрать параметры, оптимальные с точки зрения решения именно его задачи. Важным обстоятельством, усложняющим сравнение результатов сборок, является отсутствие универсальных методик оценок качества сборки.

Также достаточна распространена ситуация, когда результаты применения различных сборщиков или одного сборщика с разными параметрами существенно отличаются для одних и тех же входных данных. Обычная практика при решении задачи сборки генома заключается в запуске нескольких сборщиков с различными параметрами, а затем в выборе наилучшего варианта согласно некоторым соображениям. Однако, недавние исследования показывают, что

довольно часто одни сборщики показывают лучший результат по одному из статистических критериев в сравнении с другими программами и уступают им же по другому критерию. С большой уверенностью можно утверждать, что для каждого сборщика существуют фрагменты в геноме, с задачей сборки которых он справляется лучше остальных. В основном это объясняется деталями реализации стратегии того или иного сборщика по разрешению «проблемных» участков в геноме, например, связанных с повторами.

В силу больших объемов входных данных, которые для проектов расшифровки геномов даже небольшого размера составляют несколько десятков Гб, геномные сборщики весьма требовательны к вычислительным ресурсам. Большинству ведущих программных комплексов для сборки геномов млекопитающих необходимо либо несколько сотен Гб оперативной памяти, либо вычислительный кластер с распределенной памятью. Обычной практикой для исследовательских групп также является аренда вычислительных мощностей в облаке для сборки генома, либо использование веб-сервисов, предоставляющих интерфейс для интересующего сборщика.

Целью данной работы является разработка методов и алгоритмов, использующих ресурсы вычислительного кластера и направленных на автоматизацию процесса сборки генома *de novo* из коротких чтений. Для достижения цели работы были поставлены следующие **задачи**:

1. Провести анализ существующих программ по сборке генома из коротких чтений и выбрать наиболее подходящие для использования в описываемых методах.
2. Провести анализ существующих методов подбора оптимальных параметров для работы геномных сборщиков.
3. Разработать распределенный алгоритм объединения результатов работ различных геномных сборщиков при отсутствии референса и новый метод оценивания качества геномной сборки при отсутствии референса, соотносящийся с качеством получаемой сборки.

Основные положения, выносимые на защиту:

1. Разработан и проверен метод оценки качества геномной сборки при отсутствии референса с помощью анализа частот k-меров на основе программного средства Jellyfish.
2. Разработана система для выбора значения параметра k для сборщиков, основанных на графе де Брюйна, оптимального с точки зрения предложенного метода оценки качества геномной сборки. Исходный код доступен по ссылке: https://bitbucket.org/kromanenkov/apa.
3. На основании проведенного исследования существующих актуальных сборщиков генома и существующих методов объединения набора геномных сборок разработан и реализован распределенный алгоритм объединения результатов работ различных геномных сборщиков при отсутствии референса, основанный на кластеризации взвешенного

графа контигов. Исходный код доступен по ссылке: https://bitbucket.org/kromanenkov/gar.

Научная новизна:

1. Предложен и проверен новый метод оценивания качества геномной сборки при отсутствии референса с помощью анализа частот k-меров.
2. Разработан новый распределенный алгоритм объединения результатов работ различных геномных сборщиков при отсутствии референса, основанный на кластеризации взвешенного графа контигов.

Практическая значимость. Предложенный метод оценки качества геномной сборки при отсутствии референса с помощью анализа частот k-меров лег в основу программной системы, позволяющей запускать геномные сборщики, основанные на графе де Брюйна, и выбирать для них значение параметра k, оптимальное с точки зрения предложенного метода. Показано, что предложенный метод точнее, чем существующие аналоги, отражает референс-зависимые характеристики генома. Такая система может быть полезна в проектах сборки геномов *de novo*, когда характеристики исследуемого генома неизвестны, а стандартные метрики не всегда отражают качество получаемой сборки. Также такая система применима для, вообще говоря, оптимального выбора любых параметров, используемых сборщиком.

Предложенный алгоритм объединения различных геномных сборок при отсутствии референса был реализован в качестве программного инструмента, позволяющего решать задачу согласования геномных сборок. Данный инструмент может применяться для улучшения черновых версий сборок генома *de novo*, в силу вышеупомянутой проблемы неизвестных характеристик исследуемого генома, а также тем фактом, что различные сборки одного и того же организма могут дополнять друг друга. Предложенный метод позволяет уменьшить фрагментацию последовательностей и увеличить среднюю длину последовательности в объединении, при этом превосходя существующие аналоги по большинству референс-зависимых характеристик генома. Важно отметить, что данный программный инструмент не требует от пользователя никаких дополнительных данных, а также разнообразного программного обеспечения, установленного на целевой машине.

Методология и методы исследования. В диссертационной работе применялись методы теории графов, выделения сообществ в сетях, а также теории алгоритмов и теории вероятностей.

Апробация работы. Основные положения работы докладывались на следующих конференциях и семинарах:

- Международной научной конференции «Параллельные Вычислительные технологии» (г. Екатеринбург, 2015 г.);
- Международной конференции по биоинформатике и биомедицине The IEEE International Conference on Bioinformatics and Biomedicine (г. Вашингтон, 2015 г.);
- Международной конференции студентов, аспирантов и молодых ученых «Ломоносов-2016» (г. Москва, 2016 г.);

– Международной конференции по вычислительной молекулярной биологии «МССМВ-2017» (г. Москва, 2017 г.);
– Семинаре НИЦ КИ (г. Москва, 2016 г.);
– Семинаре ИБХ РАН (г. Москва, 2016 г.);
– Семинаре им. М.Р. Шура-Бура под руководством М.М. Горбунова-Посадова (г. Москва, 2018 г.);

Публикации. Основные результаты диссертационной работы опубликованы в 6 печатных изданиях, 2 из которых изданы в журналах, рекомендованных ВАК [1; 2], 1 входит в международную библиографическую базу Scopus [3], 3 - в тезисах докладов [4–6].

Личный вклад автора заключается в выполнении теоретических и экспериментальных исследований, изложенных в диссертационной работе, включая разработку методик, разработку и реализацию алгоритмов, анализ и оформление результатов в виде публикаций и научных докладов.

В работе [1] личный вклад автора состоит в разработке и реализации предлагаемого распределенного метода объединения геномных сборок. В работе [6] личный вклад автора состоит в разработке и проверке предложенного метода оценки качества сборки.

Содержание работы

Во **введении** обосновывается актуальность исследований, проводимых в рамках данной диссертационной работы, формулируется цель, ставятся задачи работы, излагается научная новизна и практическая значимость представляемой работы.

Первая глава посвящена описанию существующих технологий секвенирования генома, методам сборки геномов, а также основным проблемам, возникающим при этом. Технологии секвенирования генома можно разделить на три основные группы: традиционные методы (секвенирование по Сэнгеру), методы секвенирования нового поколения (в ряде источников называются методами секвенирования 2-го поколения) и секвенирование с помощью длинных ридов (в ряде источников называются методами секвенирования 3-го поколения).

Раздел 1.1 содержит краткое описание метода Сэнгера, а также сборку фрагментов генома, полученных по этой технологии. Метод Сэнгера — самый надежный метод секвенирования, однако он достаточно долог и требует значительных вложений с точки зрения инфраструктуры, реагентов, образцов ДНК и трудовых ресурсов. Общая стоимость секвенирования генома млекопитающих с использованием этой технологии в 2005 году составляла около 12 млн $. Середина 2000-х годов характеризовалась несколькими прорывными оптимизациями технологического процесса секвенирования.

Раздел 1.2 содержит описание коммерциализированных в настоящее время методов секвенирования нового поколения, а также секвенирования с помощью

длинных ридов. В разделах 1.2.1–1.2.4 описаны технологии секвенирования, относящиеся ко второму поколению: 454, Illumina, SOLiD и Ion Torrent, — приведены схемы этапов работы для ряда методов. Технологии секвенирования нового поколения позволяют получать короткие чтения с более высокой частотой ошибок, но гораздо быстрее и дешевле, чем при секвенировании по Сэнгеру. Разделы 1.2.5 и 1.2.6 содержат краткое описание технологий секвенирования третьего поколения: SMRT и MinION, — позволяющих прочитывать длинные фрагменты генома со значительно большей частотой ошибок. В разделе 1.2.7 проводится сравнение описанных методов секвенирования, приводятся диаграммы разделения рынка секвенирования и используемых в настоящее время секвенаторов. Наибольшую популярность имеет платформа Illumina, а подавляющее большинство используемых секвенаторов генерируют короткие чтения. Все перечисленные технологии не позволяют получить информацию о позиции прочитанных фрагментов в геноме. Другими словами, единственное предположение, которое можно делать относительно местоположений ридов — это то, что геном покрыт ими относительно равномерно.

В разделе 1.3 схематично описаны основные алгоритмы, используемые при сборки геномов: метод перекрытий (« overlap-layout-consensus»), поиск с хэшированием и подходы, основанные на графе де Брюйна. Метод перекрытий, как правило, применяется для сборки длинных ридов, полученных с помощью секвенирования по Сэнгеру или с помощью технологий секвенирования 3-го поколения. Однако этот метод неэффективен при работе с данными секвенаторов нового поколения, так как требует построения выравнивания для всевозможных пар ридов, имеющего квадратичную временную сложность. В силу того, что секвенаторы нового поколения генерируют несколько миллиардов ридов, для сборки данных такого рода были разработаны новые алгоритмы: менее популярный поиск с хэшированием и подход, основанный на графе де Брюйна и используемый в большинстве популярных геномных сборщиков.

Задача сборки генома из ридов может быть формализована как задача поиска наименьшей общей надстроки. Пусть имеется набор строк (ридов) $s_1, s_2, ...s_n$. Необходимо найти наименьшую строку (геном) s, такую что $\forall i \in [1,n] \quad s_i$ является подстрокой строки s. Задача поиска наименьшей общей надстроки относится к классу NP–hard задач и, значит, не имеет полиномиального алгоритма решения. Подобное представление для задачи сборки имеет следующий недостаток: в случае наличия нескольких одинаковых ридов, полученных из повторяющихся участков, наименьшая общая надстрока будет содержать каждый из этих повторов в единственном экземпляре. Поэтому в научной литературе, в основном, можно встретить две основные графовые постановки задачи сборки: граф перекрытий и граф де Брюйна, — описанию которых посвящен раздел 1.4.

В разделе 1.4.1 вводится понятие графа перекрытий, как графа, в котором каждая вершина соответствует строке s_i, а ребро — ситуации перекрытия между парой строк, и дается определение графа строк, как преобразованного графа

перекрытий. Задача сборки генома из набора строк эквивалентна нахождению обобщенного Гамильтонова пути (то есть пути, который включает каждую вершину в графе по крайней мере по одному разу) минимальной длины в графе строк. В работах других авторов показано, что эта задача также относится к классу NP–hard.

Раздел 1.4.2 содержит постановку задачи сборки генома с помощью графа де Брюйна. Предварительно определим $s[i,j]$ как подстроку s, начинающуюся в i-ой позиции и заканчивающуюся в j-ой позиции включительно. Подстрока строки длины k, называется k-мером. Пусть дано множество строк $S = \{s_1, ..., s_n\}$ над алфавитом Σ. Построим с его помощью ориентированный граф $B^k(S) = (V,E)$ следующим образом:

- V — множество вершин. V : $\{d \in \Sigma^k \mid \exists i : d$ — подстрока $s_i\}$
- E — множество ребер. E : $\{d[1..k] \to d[2..k+1] \mid d \in \Sigma^{k+1}, \exists i : d\}$ — подстрока s_i

Граф такого вида называется графом де Брюйна и обозначается $B^k(S)$.

Из этого следует, что k-меры, являющиеся подстроками из набора ридов, становятся вершинами графа де Брюйна, а $k+1$-меры — ребрами. Рид, представленный в виде последовательности $k+1$-меров, может быть рассмотрен как путь в графе де Брюйна. Соответственно, любой путь в графе де Брюйна, который содержит все риды в качестве своих подпутей, является корректной сборкой генома и называется суперпутем. Тогда задача сборки генома из набора строк S эквивалентна нахождению суперпути минимальной длины в графе $B^k(S)$. В работах других авторов показано, что задача нахождения суперпути минимальной длины в графе де Брюйна $B^k(S)$ принадлежит к классу NP–hard задач для $|\Sigma| \geq 3$ и для любого положительного k.

Таким образом, вне зависимости от выбранной постановки, точное решение невозможно получить в силу большой размерности входных данных, поэтому решается приближенная задача. Даже при рассмотрении приближенной задачи сборки генома, существует ряд проблем, значительно усложняющих ее решение. Описанные модели не учитывают наличие ошибок в ридах, сложную структуру повторов в геноме, неравномерное покрытие исследуемого генома чтениями. То есть не существует теоретической возможности решить за полиномиальное время задачу сборки в идеальной постановке, не говоря уже о сложностях, возникающих при расшифровке реальных геномов.

В разделе 1.5 перечислены основные факторы, затрудняющие задачу сборки генома. Первый из них, описанный в разделе 1.5.1, — неравномерность покрытия генома ридами. Поскольку в процессе секвенируется не одна, а несколько копий исследуемой ДНК, которые при этом фрагментируются в случайных местах, могут возникать ситуации, когда какой-либо регион генома недостаточно полно представлен в полученных чтениях. В классической работе Лэндера-Уотермана описана модель секвенированного генома, не учитывающая ошибки при прочтении ДНК. В ней предполагается, что все риды имеют

одинаковую длину, при этом координаты левых концов ридов в геноме моделируется Пуассоновским процессом. В статье Лэндера-Уотермана выводится ряд важных оценок для генома, собранного в таких идеальных условиях. Для демонстрации проблемы неоднородного покрытия генома ридами приводится пример секвенирования геномной последовательности специального вида, состоящей из пяти структурных частей: $<\alpha'\beta'\gamma\beta''\alpha''>$. Символы α и β используются для обозначения повторяющихся участков, для их различения к символу добавляется апостроф. Доказывается утверждение о том, что в случае нарушения покрытия этого генома — а именно отсутствия ридов, левые концы которых лежат в участке β', а правые в участке γ, — из оставшихся ридов, используя формализацию задачи сборки из раздела 1.4, возможно восстановить только геном с последовательностью $<\gamma\beta'\alpha'\beta''\alpha''>$.

Для вышеупомянутого утверждения согласно модели Лэндера-Уотермана оценивается вероятность нарушения покрытия, указанного в его условии, — существование в геноме отрезка длины L, не содержащего ни одного левого конца рида. Доказывается утверждение о том, что эта вероятность равна $e^{-\frac{NL}{G}}$, где N — число ридов, L — длина рида, а G — длина генома. В конце раздела 1.5.1 отмечается, что координаты левых концов ридов далеко не всегда подчиняются Пуассоновскому распределению, а модель Лэндера-Уотермана является достаточно упрощенной, поскольку не учитывает различную длину ридов и ошибки секвенирования.

Раздел 1.5.2 содержит описание следующего фактора, затрудняющего задачу сборки геномов, — наличие в них повторяющихся участков. Повторы могут иметь различную структуру, длину и встречаться в любых местах молекулы ДНК. Наличие повторов может приводить к невозможности однозначного восстановления исходной последовательности даже при условии равномерного покрытия генома ридами. Для демонстрации этого тезиса доказывается следующее утверждение. Пусть секвенируется геномная последовательность специального вида: $<\alpha'\beta\alpha''\gamma\alpha'''>$. Символы α и β используются для обозначения повторяющихся участков, для их различения к символу добавляется апостроф. Тогда из множества ридов, используя формализацию задачи сборки из раздела 1.4, может быть восстановлен как исходный геном, так и геном с последовательностью $<\alpha'\gamma\alpha''\beta\alpha'''>$ и никакой другой. Отметим, что особенности разрешения характерных топологических участков в графах строк, либо в графах де Брюйна, вызванных наличием повторов в геноме, — основная причина того, что результаты применения различных сборщиков или одного сборщика с разными параметрами существенно отличаются для одних и тех же входных данных.

В разделе 1.5.3 показано влияние ошибок секвенирования на задачу сборки генома. Частота возникновения ошибок для секвенаторов нового поколения составляет около 1 процента, однако это компенсируется тем, что каждый нуклеотид, как правило, встречается в нескольких ридах. На этом факте основано

большинство алгоритмов коррекции ридов. В разделе приводятся оценки, полученные в работах других авторов, для вероятности неисправления ошибок секвенирования согласно принципу консенсуса и для длины рида, необходимой для успешного восстановления исходной последовательности из ридов, содержащих ошибки. Представлен пример неверной сборки генома из четырех ридов из-за ошибочного прочтения одного из нуклеотидов.

Вторая глава посвящена методике оценивания качества геномной сборки при отсутствии референса. Из-за влияния факторов, перечисленных в разделе 1.5, результатом сборки генома *de novo* практически никогда не являются непосредственно последовательности хромосом, так как в ряде случаев не удается решить даже приближенную задачу восстановления последовательностей. На выходе сборщика обычно получается набор непрерывных, возможно перекрывающихся, фрагментов исследуемой ДНК — контигов. К настоящему времени разработано множество методик сравнения наборов контигов между собой, самые распространенные из которых приведены в разделе 2.1. Их можно разделить на две основные категории: не требующие уже собранной каким-либо другим способом исследуемой последовательности («референсная последовательность») и требующие наличия референса. Отметим, что в случае сборки генома *de novo* возможно использование только безреференсных методик, при этом в настоящее время ни одна из них не является общепризнанной.

В разделе 2.1.1 перечислены безреференсные методики оценки качества геномной сборки: число контигов, суммарная длина контигов и N50. Значение метрики N50 — это максимально возможная длина контига L, такая, что контиги с длиной $\geq L$ составляют по крайней мере половину от суммарной длины контигов. Данная методика считается одной из основных при сравнении геномных сборок *de novo* и призвана отразить баланс между числом контигов, их средней и суммарной длиной. Большее значение N50 соответствует лучшей сборке и приблизительно соответствует мере фрагментированности набора контигов. Однако метрика N50 не всегда объективно отражает качество сборки. Для демонстрации этого тезиса рассматриваются два множества контигов: первое более фрагментированное чем второе, при этом значение N50 для второго набора выше чем для первого. Доказывается утверждение о том, что если второе множество контигов дополнить короткими последовательностями, суммарная длина которых превосходит размер содержащихся в нем элементов, то значение N50 для модифицированного таким образом множества будет меньше, чем у первого множества.

В разделе 2.1.2 перечислены методики, использующие референсный геном. Кажущаяся на первый взгляд бессмысленной задача сборки уже восстановленной до этого последовательности имеет множество практических приложений. Геномы многих организмов, принадлежащих к одному виду, не являются точной копией друг друга, и именно участки отличия между ними представляют интерес для изучения. Полная сборка референсной последовательности обычно не обходится без секвенирования по Сэнгеру, требующего

значительных финансовых и временных ресурсов, поэтому геномов, для которых существует референс, пока не так много. Тем не менее, в случае наличия такой последовательности нижеперечисленные метрики помогают оценить корректность полученной сборки.

Первая из описанных в этом разделе метрик — ошибки сборки — основана на подсчете неверно картирующихся на референс контигов. Значение другой, NG50, вычисляется как максимально возможная длина контига L, такая, что контиги с длиной $\geq L$ составляют по крайней мере половину от длины референсного генома, а не суммарной длины контигов, как в случае N50. Метрика NGA50, которую еще называют скорректированным N50, является комбинацией NG50 и числа ошибок сборки и рассчитывается следующим образом. Прежде всего производится выравнивание контигов на референс. Если последовательность контига содержит ошибку сборки одного из вышеописанных типов, то в этой точке он разбивается на два отдельных блока, для которых снова проводится процедура картирования на референс. Если какой-то из получившихся блоков невозможно выравнять на референс, то он исключается из рассмотрения. Для полученного таким образом набора контигов рассчитывается стандартная метрика NG50. Значение NGA50 всегда меньше или равно значению NG50 и, по сути, штрафует сборщик за склеивание участков, не следующих друг за другом в геноме. Более высокое значение NGA50 обычно соответствует лучшей сборке. Последняя из упоминаемых в этом разделе метрик — длиннейший непрерывный участок. Значение этой метрики — длина самого протяженного блока, образованного в результате процедуры разбиения контигов для вычисления NGA50. По сути, это скорректированная длина самого протяженного контига.

Раздел 2.2 посвящен описанию предложенной методики оценивания качества геномной сборки на основе встречаемости k-меров. Практически любой геном содержит повторяющиеся участки, однако, начиная с определенного значения k, k-меры в некотором роде однозначно идентифицируют его; если посчитать числа встречаемости k-меров для достаточно большого k (ограниченного при этом сверху длиной рида) у генома относительно небольшого размера, получится, что большая часть из них встречается в геноме в единственном экземпляре. Отметим, что *de novo* сборка больших геномов сложной структуры малоосмысленна, и обычно подразумевает либо наличие референса, либо огромное число ридов, полученных с помощью различных технологий секвенирования (например, Illumina и PacBio).

В разделе 2.2.1 показывается, что с увеличением значения k вероятность многократного содержания k-мера в геноме стремится к 0. Исходя из предположения, что геном — это случайная строка длины G, а вероятности появления символов нуклеотидов в строке не зависят от позиции и составляют 0.25, показано, что вероятность встретить случайную строку длины L в последовательности генома хотя бы один раз составляет $1 - (1 - \frac{1}{4^L})^{G-L+1}$. Данное приближение является достаточно грубым и не учитывает структуру генома и строки. Тем

не менее, этот способ вполне подходит для приближенной оценки такой длины подстроки, при которой вероятность её встречи в геноме достаточно мала. Считая, что последовательность генома случайна, в качестве строки из вышеупомянутого утверждения можно выбрать его собственный k-мер. Как следует из утверждения, для достаточно большого k вероятность того, что k-мер случайно совпадет с какой-либо подстрокой генома близка к нулю. К примеру, в геноме микроспоридии Encephalitozoon cuniculi fungus 2295551 из 2373670 21-меров встречаются в единственном экземпляре.

Раздел 2.2.2 посвящен описанию сторонних программ для подсчета числа k-меров в последовательностях. Излагаются аргументы в пользу выбора программного средства Jellyfish. В разделе 2.2.3 изложена предлагаемая методика оценки качества геномной сборки на основе частот k-меров. Основная идея методики заключается в установлении соответствия между уникальными k-мерами в собранном геноме и k-мерами в ридах.

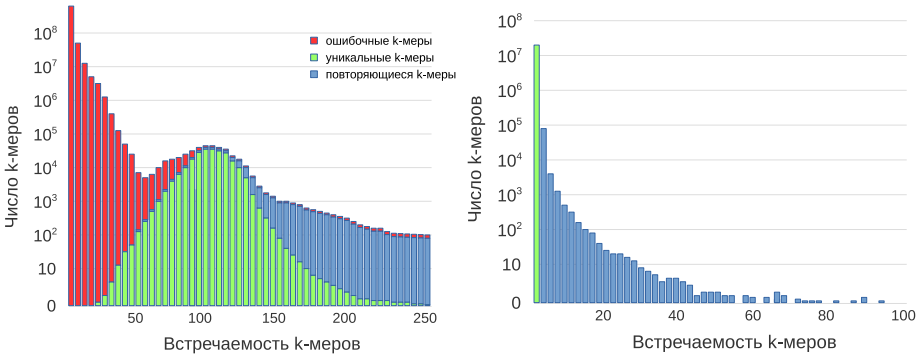
Для этого строится гистограмма встречаемости k-меров в коротких фрагментах, полученных в результате секвенирования. Пример такой гистограммы для организма Encephalitozoon cuniculi fungus при $k = 21$ изображен на рисунке 2а. По оси X отложены числа встречаемости k-меров в наборе чтений, а по оси Y отложено количество всевозможных k-меров с данной встречаемостью. Видно, что гистограмма имеет два пика: первый связан с ошибками чтения генома, а второй соответствует уникальным k-мерам в исходном геноме. Большое количество ошибочных (ложных) k-меров в наборе чтений объясняется тем, что ошибка в прочтении даже одного символа (например, ошибка замены) приводит к потенциальному возникновению нового k-мера, до этого не присутствующего в этом множестве.

Число встречаемости у второго пика (около 116) обусловлено покрытием при чтении генома: количеством прочтений каждого символа. На рисунке 2б изображена гистограмма встречаемости 21-меров собранного генома Encephalitozoon cuniculi fungus с помощью программы Velvet. Из-за того, что каждый участок генома прочитывается несколько раз, каждый уникальный k-мер из собранного генома встречается во множестве k-меров коротких чтений около 116 раз.

Предлагается вычислять долю различных k-меров, взятых из некоторой окрестности второго пика на гистограмме встречаемости k-меров в чтениях, среди уникальных k-меров для собранного генома по следующей формуле:

$$Q = \frac{\sum\limits_{i=1}^{|K_{uniq\_reads}|} [k^r(i) \in K_1^g]}{|K_{uniq\_reads}|}, \qquad (1)$$

где $K_1^g = \{k^g : k^g \in K^g, abundance(k^g) = 1\}$, $K_i^r = \{k^r : k^r \in K^r, abundance(k^r) = i\}$, $K_{uniq\_reads} = \bigcup_{i=a_{p\_l}}^{a_{p\_r}} K_i^r$, $[a_{p\_l}; a_{p\_r}]$ - некоторая окрестность из второго пика на гистограмме распределения числа встречаемости k-меров коротких чтений, $abundance(k)$ - число встречаемости k-мера k,



а) б)

Рис. 2 — Число встречаемости k-меров ($k = 21$) в наборе чтений (а) и собранном геноме (б) Encephalitozoon cuniculi fungus. Логарифмическая шкала.

K^r - множество всех k-меров коротких чтений, K^g - множество всех k-меров собранного генома.

Чем больше значение Q, тем лучше полученная сборка соотносится с результатами секвенирования. Таким образом, предложенная методика устанавливает соответствие между набором коротких чтений, полученных в результате секвенирования, и собранным геномом, позволяя более точно оценивать результат геномной сборки.

Сама процедура сравнения сборок согласно предложенной методике выглядит следующим образом:

1. Построение гистограммы встречаемости k-меров для ридов, полученных при секвенировании исследуемого генома. Так как молекула ДНК содержит две комплементарные цепочки, вводится понятие канонического k-мера: этот такой k-мер из пары k-мер и комплементарный k-мер, который меньше лексикографически. При подсчете чисел встречаемости k-меров сохраняются последовательности только канонических k-меров, при этом числа встречаемости для k-мера и его комплемента суммируются.
2. Выбор некоторой окрестности пика уникальных k-меров на гистограмме встречаемости k-меров в ридах.
3. Построение гистограммы встречаемости k-меров для каждой из полученных сборок.
4. Подсчет значения Q. Каждый k-мер из множества уникальных k-меров ридов проверяется на вхождение во множество уникальных k-меров собранного генома. Для этого подсчитываются числа встречаемости k-мера и его комплемента во множестве k-меров рассматриваемой сборки. Если сумма чисел их встречаемости не равна единице, то такой

k-мер отбрасывается. Если же k-мер и его комплемент в сумме встретились ровно один раз, то считается, что данный k-мер принадлежит множеству уникальных k-меров собранного генома.

5. Выбор сборки с максимальным значением Q в качестве наилучшей.

Значение Q зависит от размера выбранной окрестности пика уникальных k-меров на гистограмме встречаемости для ридов: как показали эксперименты, оно уменьшается с увеличением размера окрестности. Это объясняется тем, что при расширении границ окрестности знаменатель дроби в формуле 1 увеличивается на число k-меров, попавших в расширенную окрестность, в то время как числитель увеличивается только на число добавленных k-меров, принадлежащих множеству уникальных k-меров собранного генома.

Поскольку значение Q зависит от размера окрестности пика уникальных k-меров на гистограмме встречаемости для ридов, важно корректно выбрать ее границы. В разделе 2.2.4 описан способ выбора такой окрестности:

1. На гистограмме встречаемости k-меров в ридах ищется второй пик, соответствующий уникальным k-мерам в исходном геноме, и локальный минимум между двумя пиками. Пусть число встречаемости для пика уникальных k-меров равно p, а для локального минимума m.

2. Значение a_{p_r} устанавливается равным $2(p-10)-\alpha : \alpha \in [0;8], a_{p\_r} \vdots 10$. Выбор правой границы объясняется следующим образом: если уникальные k-меры из собранного генома встречаются во множестве k-меров ридов около p раз, значит k-меры, содержащиеся в собранном геноме два раза, будут встречаться во множестве k-меров ридов около $2p$ раз и так далее. То есть, правая граница выбирается таким образом, чтобы исключить из окрестности неуникальные k-меры собранного генома.
3. Для выбора левой границы приближенно оценивается размер исследуемого генома без учета повторов с помощью формулы, описанной в работе Li и Waterman:

$$G = \frac{S(L-k+1)}{LM},$$

где S - общий размер ридов, L - длина рида, k - размер k-мера, M - покрытие k-меров, которое определяется как число встречаемости для пика уникальных k-меров, то есть $M = p$.

4. Выбирается сборка, общий размер контигов которой ближе всего к оцененному размеру генома, затем из сборки исключаются контиги короче 500 символов, а для оставшихся контигов подсчитывается число уникальных k-меров. Обозначим это число как $K_1'^g$, тогда левая граница вычисляется следующим образом:

$$a_{p\_l} = \max \begin{cases} \max\limits_x : |\bigcup_{i=x}^{a_{p\_r}} K_i^r| \geq 0.85|K_1'^g| \\ m-\alpha : \alpha \in [1;10], (m-\alpha) \vdots 10 \end{cases}.$$

То есть требуется, чтобы количество уникальных k-меров ридов составляло по крайней мере 85% от количества уникальных k-меров в контигах длиною $\geq$ 500 символов для сборки, размер которой ближе всего к размеру оцененного генома, и при этом левая граница не сильно отличалась от числа встречаемости локального минимума. Это ограничение необходимо, чтобы исключить из окрестности ошибочные k-меры.

В разделе 2.3 описана программная реализация предложенного метода, изложенного в разделе 2.2.3. Построение гистограммы встречаемости k-меров, сериализация k-меров, а также вычисление значения Q проводилось с использованием программы Jellyfish. Раздел 2.4 содержит результаты проверки предложенной методики. Для анализа корректности работы предложенной методики была проведена серия вычислительных экспериментов, в ходе которых производилась сборка генома микроспоридии Encephalitozoon cuniculi fungus четырьмя различными сборщиками: ABYSS, Ray, SOAPdenovo и Velvet. Отобранные в этой работе сборщики объединяют следующие критерии: они фигурируют в статьях, посвященных сравнительному анализу качества геномных сборок, используют концепцию графа де Брюйна, при этом не используя более одного значения k для сборки за раз, имеют открытый исходный код, и им для работы достаточно одного набора ридов.

Сборка генома микроспоридии производилась для пяти различных значений параметра k: 25, 33, 43, 51, 59. Выбор этих значений обусловлен следующими соображениями: при сборке генома *de novo* минимальное значение k обычно устанавливается равным 21 (например, в случае сборщика SPAdes), исходя из рассуждений, аналогичным утверждению, доказанному в разделе 2.2.1. Верхняя граница параметра, как правило, не превосходит 60–70 % от длины рида, в данном случае равной 100. Промежуточные значения призваны продемонстрировать необходимость выбора для каждого сборщика собственного значения k, обеспечивающего наилучшее качество сборки.

Так как для данного генома существует референс, то оценка качества работы предложенной методики выглядела следующим образом: каждый сборщик запускался с пятью различными значениями параметра k, для полученных сборок производился сбор статистик, описанных в разделе 2.1, с помощью программы Quast и подсчет значений Q, и пять сборок упорядочивались согласно подсчитанным значениям. В качестве меры «правильности» полученных сборок была выбрана референс-зависимая метрика NGA50, позволяющая достаточно точно судить о качестве геномных сборок и описанная в пункте 2.1.2. Назовем набор сборок, упорядоченный в соответствии с этой метрикой, каноническим. Набор сборок, упорядоченный в соответствии со значением Q, сравнивался с каноническим набором. Ясно, что чем меньше различий существует между двумя наборами, тем лучше предложенная методика отражает качество геномной сборки, по крайней мере в терминах метрики NGA50.

Разделы 2.4.1–2.4.4 содержат анализ результатов работы четырех вышеупомянутых сборщиков. В таблицах 1–4 показаны результаты сравнения полученных геномных сборок.

Таблица 1 — Результаты сравнения различных сборок Encephalitozoon cuniculi fungus программой Velvet

k	N50	ДНУ	Число контигов	NGA50	Длина контигов (в Mb)	Q
25	1175	39958	15927	11316	17.7	0.934086
33	928	65298	14977	21452	14.5	0.938702
43	937	73538	8093	18836	8.3	0.937946
51	4091	39729	2610	9237	3.7	0.92775
59	3339	17502	1124	3189	2.4	0.905697

Таблица 2 — Результаты сравнения различных сборок Encephalitozoon cuniculi fungus программой SOAPdenovo

k	N50	ДНУ	Число контигов	NGA50	Длина контигов (в Mb)	Q
25	1101	3535	16920	644	17.4	0.639442
33	932	13671	12249	2578	11.5	0.898704
43	1383	39438	3816	14839	4.7	0.936651
51	7480	43171	1245	9248	2.8	0.92729
59	3412	17498	1086	3183	2.3	0.901179

Таблица 3 — Результаты сравнения различных сборок Encephalitozoon cuniculi fungus программой Abyss

k	N50	ДНУ	Число контигов	NGA50	Длина контигов (в Mb)	Q
25	1495	71513	14981	28200	19.7	0.932986
33	1110	142490	12913	55107	14.3	0.940028
43	1082	177241	4939	63324	5.8	0.940482
51	41991	149473	1241	55270	3.1	0.934639
59	20966	76543	488	20966	2.5	0.926928

Таблица 4 — Результаты сравнения различных сборок Encephalitozoon cuniculi fungus программой Ray

k	N50	ДНУ	Число контигов	NGA50	Длина контигов (в Mb)	Q
25	950	194751	8820	78910	9.3	0.895447
33	5780	108186	3274	42981	4.6	0.927833
43	14182	54559	758	15087	2.7	0.92385
51	6909	25103	339	4018	1.6	0.603678
59	2827	14221	419	-	0.9	0.364621

В случае сборщиков Velvet и Soap последовательность значений k, упорядоченная в порядке убывания метрики NGA50, полностью совпадает с последовательностью k, упорядоченной в порядке убывания значений Q. Для сборщика Abyss последовательность значений k, упорядоченная по убыванию значения Q, отличается от канонического набора одной перестановкой, однако несмотря на то, что значение NGA50 для $k = 51$ выше, чем для $k = 33$, NGAx для $k = 33$ больше NGAx для $k = 51$ для бóльшего количества x и с бóльшей разницей. Поэтому при упорядочивании последовательности значений k сборщика Abyss в порядке убывания метрики NGAx, она полностью согласуется с последовательностью, получаемой в результате анализа значений Q. В случае сборщика Ray эта последовательность также отличается от канонического набора одной перестановкой. Это объясняется особенностями сборки, построенной для $k = 25$. Для демонстрации этого факта вводится вспомогательная величина $\widetilde{Q}$:

$$\widetilde{Q} = \frac{\sum_{i=1}^{K_{uniq\_reads}} [k^r(i) \in \widetilde{K}_n^g]}{|K_{uniq\_reads}|}, \quad (2)$$

где $\widetilde{K}_n^g = \{k^g : k^g \in K^g, abundance(k^g) \geq 1\}$.

Q всегда не превосходит $\widetilde{Q}$, однако слишком большая разница их значений свидетельствует о потенциальных проблемах сборки. Показано, что относительно низкое значение Q при $k = 25$, в сравнении с другими значениями k, объясняется аномально высоким показателем дублирования у этой сборки. Это проявляется и в заметном скачке разницы между Q и $\widetilde{Q}$ при $k = 25$, что вызывает определенные вопросы к качеству подобной сборки. Показано, что в данном случае значение Q точнее отражает качество сравниваемых сборок, чем NGAx и $\widetilde{Q}$. При этом для всех сборщиков не наблюдается корреляция между стандартными безреференсными метриками и значениями NGA50.

В разделе 2.5 рассматриваются существующие аналоги предложенного метода: KmerGenie и ALE. Программное средство KmerGenie предназначено для автоматического выбора оптимального значения k. Принцип его работы основан на анализе гистограммы встречаемости k-меров в наборах коротких чтений

для разных значений k, при этом в отличие от предложенного метода никак не оценивается соответствие между k-мерами в ридах и сборках. К сожалению, провести сравнение KmerGenie и предлагаемого метода не удалось, в силу того, что при запуске KmerGenie на чтениях генома Encephalitozoon cuniculi fungus программа аварийно завершила свою работу.

ALE — это программа для качественной оценки сборки геномной последовательности на основе принципа максимального правдоподобия. В качестве ответа она выдает логарифм вероятности того, что сборка является верной при наличии заданного набора чтений. Для сравнения ALE и предложенного метода для каждой из полученных сборок был посчитан ее логарифм вероятности с помощью программы ALE. Результаты этой оценки приведены в таблице 5. К сожалению, на ряде данных ALE аварийно завершила свою работу, такие случаи обозначается в таблице с помощью N/A.

Таблица 5 — Результаты оценки различных сборок программой ALE

k	Velvet	Soap	Abyss	Ray
25	-12004×10^5	-12121×10^5	-12075×10^5	-11646×10^5
33	-11912×10^5	N/A	N/A	-11488×10^5
43	N/A	N/A	N/A	N/A
51	N/A	-11454×10^5	N/A	N/A
59	-11448×10^5	N/A	N/A	-11576×10^5

Несмотря на то, что во многих случаях программа не смогла выдать результат, полученные цифры свидетельствуют об отсутствии связи между логарифмом вероятности того, что сборка является верной, и значением метрики NGAx. Похоже, что единственный показатель, с которым коррелирует оценка ALE — это суммарная длина контигов.

Раздел 2.6 содержит результаты анализа предложенной методики оценки качества геномной сборки при отсутствии референса с помощью анализа частот k-меров. Было установлено, что в большинстве случаев последовательность сборок, упорядоченная в порядке убывания значения NGAx, полностью согласуется с последовательностью сборок, упорядоченной в порядке убывания значения Q. Таким образом, показано, что предложенная методика коррелирует с референс-зависимыми метриками и позволяет корректно определять лучшую сборку. При этом не была выявлена взаимосвязь между качеством сборки и стандартными метриками.

Третья глава посвящена задаче объединения результатов работ различных геномных сборщиков. Эта задача, аналогично задаче сборки генома, может быть формализована как поиск наименьшей общей надстроки. Однако, по сравнению с задачей сборки, у нее есть две особенности: повторяемость исходного генома в наборах контигов, приводящая к высокому показателю дублирования

у итоговой сборки, и потенциальное возникновение ошибочных последовательностей, возникающее при склеивании контигов, не следующих друг за другом в геноме.

В разделе 3.1 обосновывается актуальность данной задачи, перечисляются существующие программы, позволяющие проводить объединение трех и более наборов контигов: Mix и CISA, отмечаются их недостатки. Проблемы при использовании этих методов, в основном, связаны с наличием повторяющихся участков в геноме. В этом случае граф строк, определенный в разделе 1.4.1 (в данном случае в качестве строк выступают контиги), может содержать ложные ребра. Ложными называются ребра, которые соединяют контиги, не следующие друг за другом в референсном геноме, но имеющие перекрытие из-за ошибок, допущенных на этапе сборки, или из-за наличия у них одинаковых повторяющихся участков. На рисунке 3 представлен референсный геном, содержащий три повторяющихся участка, обозначенных зелеными отрезками. Пусть выполняется объединение двух сборок, содержащих два и три контига соответственно. Из-за наличия повторов граф контигов содержит ложное ребро: $1.3 \to 2.1$. Отметим, что ребро между вершинами 2.2 и 1.3 не является ложным: несмотря на то, что контиги, соответствующие этим ребрам, не следуют друг за другом в референсной последовательности, при их объединении получается корректная подстрока референса.

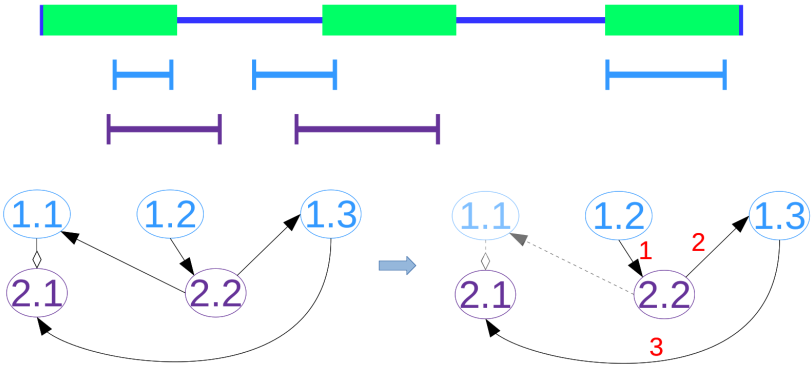


Рис. 3 — Пример графа строк, содержащего ложное ребро.

Обобщенный Гамильтонов путь в этом графе выглядит так: 1.2; 2.2; 1.3; 2.1, и контиг, соответствующий этому объединению, будет содержать ошибку типа «*Relocation*», описанную в разделе 2.1.2.

Раздел 3.2 содержит описание предлагаемого алгоритма объединения различных геномных сборок при отсутствии референса. Суть предлагаемого алгоритма состоит в построении графа контигов, его кластеризации и объединении контигов в найденных областях. Первый шаг алгоритма, описанный в разделе

3.2.1, заключается в построении попарного выравнивания контигов, выполняемого с помощью программы Megablast. Предложенный метод разделяет варианты различного расположения контигов: обычное концевое перекрытие, концевое перекрытие последовательностей из разных цепей ДНК, полное совпадение и вложенность. Остальные варианты взаимного расположения контигов игнорируются. Для каждого случая перекрытия Megablast собирает статистику выравнивания. В силу того, что каждый контиг будет многократно участвовать в выравнивании, все множества контигов предварительно индексируются. Ввиду большой размерности матрицы попарного выравнивания контигов для пары наборов, она хранится в разреженном виде. Реализация разреженной матрицы выполнена с помощью хэш-таблицы, ключом в которой служит пара индексов контигов в наборах.

В разделе 3.2.2 описан второй шаг алгоритма — построение графа контигов и его кластеризация. На основе матриц попарного выравнивания контигов производится построение взвешенного неориентированного графа контигов. Вершины графа соответствуют контигам, а ребра — их перекрытию. Вес ребра равен *score* соответствующего парного выравнивания и, по сути, характеризует «перспективность» объединения конкретной пары последовательностей. С каждым ребром ассоциируется дополнительная информация о выравнивании: статистика выравнивания, описанная в разделе 3.2.1, и тип перекрытия.

Затем производится кластеризация построенного графа с помощью LabelRank-подобного алгоритма на непересекающиеся области так, чтобы максимизировать вес внутри каждой области и минимизировать вес ребер между областями. Процесс разбиения графа позволяет уменьшить число перекрытий, получающихся в результате ошибки сборки или из-за наличия в геноме длинных повторов. Алгоритм LabelRank предназначен для выделения сообществ в невзвешенных графах, однако довольно легко может быть обобщен на случай взвешенных графов. Для работы LabelRank ставит в соответствие каждой вершине множество пар <метка; вероятность>. Кластеризация заключается в изменении этого множества для каждой вершины, при этом в результате работы алгоритма вершины, у которых значения меток с максимальной вероятностью совпадают, попадают в один и тот же кластер. Отметим, что в модифицированной версии алгоритма в граф контигов для каждой вершины, имеющей соседей, добавляется петля, взвешенная усредненным весом ее ребер.

Раздел 3.2.3 содержит описание третьего шага алгоритма. На этом этапе происходит объединение контигов, попавших в один кластер, жадным алгоритмом. Для этого в кластере выбираются контиги, не являющиеся подпоследовательностями никаких других контигов (в том числе и из других кластеров), называемые затравками. Затравки расширяются в левую и правую стороны за счет контигов, находящихся в том же кластере и имеющих с затравкой общее ребро. Если таких вершин несколько, то выбирается та, перекрытие которой с текущим контигом имеет максимальный *score*. Выбранная вершина становится

текущей и процесс итеративно повторяется. Расширение затравки прекращается, когда множество вершин, за счет которых можно было бы увеличить длину текущего контига, оказывается пустым.

Одна из проблем, возникающая при объединение сборок, — это появление дубликатов, поскольку значительная часть генома дублируется в наборах контигах. Для решения этой проблемы используется следующий подход: для каждого кластера подсчитывается суммарная длина контигов, вложенных в последовательности из этого кластера, называемая числом вложенности. Последовательность кластеров упорядочивается по убыванию числа вложенности, при этом при присоединении очередного контига к затравке дерево вложенных в него контигов рекурсивно помечается как уже использованное. Использованные контиги больше не могут участвовать в объединении. Используемая концепция затравок в кластерах позволяет уменьшить показатель дублирования в объединенном наборе сборок, сохранив при этом высокий показатель покрытия генома.

На заключительном этапе работы алгоритма, описанном в разделе 3.2.4, проводится корректировка объединенных последовательностей. Для этого после объединения последовательностей в кластерах выполняется удаление концевых частей у контигов, имеющих перекрытие, но не попавших в один кластер. В противном случае область перекрытия будет присутствовать в результирующем наборе два раза, что повлечет за собой увеличение показателя дублирования.

Для борьбы с этим эффектом формируется два множества: $leftEnds$ и $rightEnds$. После того, как объединение последовательностей в кластере завершено, контиги, отвечающие левому и правому концу объединения, попадают в соответствующие множества. При объединении контигов в кластере проверяется: не перекрывается ли контиг левого конца объединения с каким-либо контигом из $rightEnds$? В случае положительного ответа на этот вопрос, данный участок перекрытия удаляется из контига левого конца объединения. Аналогичная процедура повторяется и для контига правого конца объединения и множества $leftEnds$.

На рисунке 4 показан пример работы предложенного метода объединения результатов работ геномных сборок для 3 наборов контигов.

В данном примере в результате кластеризации контиги будут разбиты на три группы. При этом в контиг 1.2 вложены контиги 2.2 и 2.3. Это значит, что число вложенности для кластера 1 равно двум, а для кластеров 2 и 3 оно равно нулю. Таким образом, объединение контигов будет начато с кластера 1. Контиг 1.2 входит в объединение контигов в данном кластере (более того, он является затравкой, как самый длинный), поэтому контиги 2.2 и 2.3, которые он покрывает, будут помечены как использованные. Поскольку в кластере 2 не осталось неиспользованных контигов, объединение последовательностей в нем не производится. Контиг 3.4, являющийся левым концом объединения контигов в кластере 3, перекрывается с контигом 1.2, входящим во множество $rightEnds$, поэтому участок перекрытия между ними удаляется из контига 3.4.

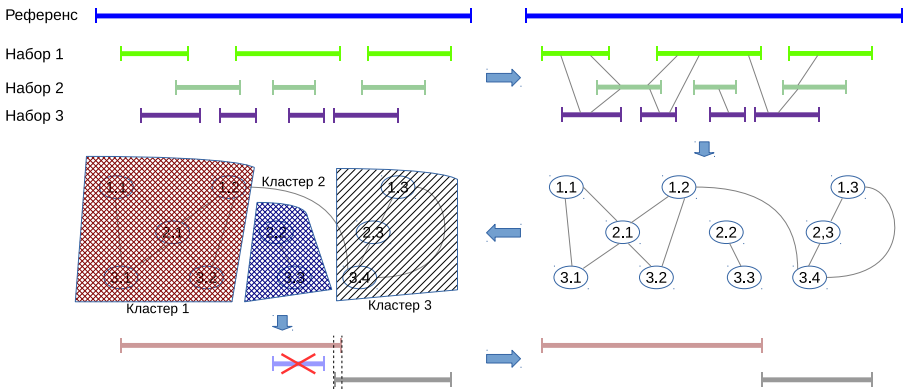


Рис. 4 — Пример работы предложенного метода для 3 наборов контигов.

В разделе 3.2.5 дается оценка временной сложности предложенного метода. Показывается, что самым вычислительно-сложным этапом метода в случае разреженного графа контигов является попарное выравнивание контигов.

Раздел 3.2.6 содержит сведения о распараллеливании предложенного метода. Исходя из оценки временной сложности алгоритма, для распараллеливания был выбран этап попарного выравнивания контигов библиотекой Megablast. Для организации параллельных вычислений и пересылок данных между процессами используется библиотека MPI.

В разделе 3.3 описывается процедура проверки предложенного метода объединения различных геномных сборок. Для этого было отобрано семь геномных сборщиков, представленных в разделе 3.3.1, и три эукариотических геномах, характеристики которых приведены в разделе 3.3.2. Для этих геномов доступна референсная последовательность, что позволяет оценить качество проведенного объединения. Также предложенный метод был протестирован на наборе сгенерированных контигов. Симулированные контиги получаются в результате случайного разбиения референсной последовательности на множество фрагментов предопределенной длины и удаления у них концевых частей. Наборы сгенерированных контигов также были использованы для оценки масштабируемости предложенного метода объединения результатов геномных сборщиков.

Раздел 3.3.3 содержит результаты анализа работы предложенного метода, а также его сравнение с существующими аналогами. В таблицах 6, 7 и 8 приведены результаты сравнения трех методов объединения, а также характеристики исходных геномных сборок. Для сравниваемых методов объединения жирным шрифтом выделены лучшие значения для каждой из рассматриваемых метрик.

В таблице 9 приведены результаты работы метода по объединению десяти сгенерированных наборов контигов на основе генома Drosophila melanogaster.

Таблица 6 — Результаты работы по объединению набора контигов Encephalitozoon cuniculi fungus

Сборщик	**Ошибки сборки**	**Число контигов (≥ 500 символов)**	**NGA50**	**Показатель дублирования (%)**	**Покрытие генома (%)**
A5	7	10737	134246	0.2	90.965
Abyss	5	549	8767	0.7	90.971
Ray	5	650	10262	0.9	91.082
SOAPdenovo	1	1617	1295	1.9	81.646
Spades	4	14929	134106	0.2	90.817
IDBA	4	1240	133909	0.2	90.965
Velvet	1	1111	3189	1.3	88.492
Mix	**14**	1635	133909	10.4	91.635
CISA	17	**1446**	136410	2.1	92.341
Предложенный метод	17	6787	**136670**	**2**	**92.358**

Таблица 7 — Результаты работы по объединению набора контигов Candida parapsilosis

Сборщик	**Ошибки сборки**	**Число контигов (≥ 500 символов)**	**NGA50**	**Показатель дублирования (%)**	**Покрытие генома (%)**
Abyss	13	346	104678	2.2	99.406
Ray	13	1092	24137	0.8	97.490
SOAPdenovo	5	1533	16128	0.3	97.353
IDBA	20	757	38332	0.2	97.596
Velvet	4	1421	17131	0.2	97.029
Mix	47	**297**	110568	5	99.246
CISA	26	323	95331	3.9	99.249
Предложенный метод	**25**	318	**110597**	**3.8**	**99.330**

Таблица 8 — Результаты работы по объединению набора контигов Saccharomyces cerevisiae S288

Сборщик	**Ошибки сборки**	**Число контигов (≥ 500 символов)**	**NGA50**	**Показатель дублирования (%)**	**Покрытие генома (%)**
Abyss	58	757	25177	0.3	92.633
Ray	32	358	13033	0.5	57.669
SOAPdenovo	64	1633	15771	0.6	92.813
Spades	50	2584	73166	0.3	93.703
IDBA	58	1708	45419	0.3	93.480
Velvet	62	1011	18811	0.4	89.529
Mix	**55**	2515	73233	**0.5**	93.654
CISA	59	1869	74268	0.9	93.752
Предложенный метод	58	**287**	**79654**	2.2	**93.777**

Таблица 9 — Результаты работы по объединению сгенерированных наборов контигов

Номер сборки	**Ошибки сборки**	**Число контигов (≥ 500 символов)**	**NGA50**	**Показатель дублирования (%)**	**Покрытие генома (%)**
Набор 1	0	2330	74029	0.3	89.212
Набор 2	0	2317	76460	0.3	89.299
Набор 3	0	2340	73153	0.3	89.183
Набор 4	0	2333	74067	0.3	89.216
Набор 5	0	2322	74866	0.3	89.295
Набор 6	0	2343	75664	0.3	89.449
Набор 7	0	2324	75957	0.3	89.49
Набор 8	0	2327	76581	0.3	89.461
Набор 9	0	2327	74866	0.3	89.246
Набор 10	0	2315	77417	0.3	89.511
Предложенный метод	30	2186	96615	0.5	96.589

Существующие аналоги в силу своей ресуркоемкости не запускались на данном наборе данных и сравнение с ними не проводилось.

В результате данного сравнения было продемонстрировано, что предложенный метод успешно справился с основной задачей согласования геномных сборок: уменьшение фрагментации последовательностей и увеличение средней длины последовательности в объединении. Показано, что разработанный метод превосходит Mix и CISA по большинству референс-зависимых метрик генома.

В разделе 3.4 содержится описание интеграции метода в качестве модели информационно-вычислительной среды Mathcell. Предоставление веб-интерфейса к предложенному методу позволяет использовать его для улучшения черновых версий сборок генома *de novo* особенно в ситуациях, когда целевой геном неизвестен. Запуск модели на счет с использованием мощностей ИМПБ РАН избавляет пользователя от необходимости установки разнообразных библиотек, необходимых для работы программной реализации.

В **заключении** приведены основные результаты работы, которые заключаются в следующем:

1. Разработан и проверен метод оценки качества геномной сборки при отсутствии референса с помощью анализа частот k-меров на основе программного средства Jellyfish. Разработанный метод в большинстве случаев позволяет корректно определить лучшую сборку с точки зрения референс-зависимых метрик. Показано, что предложенный метод точнее, чем существующие аналоги, отражает референс-зависимые характеристики генома.
2. Для проверки метода оценки качества геномной сборки была разработана система для выбора значения параметра k для сборщиков, основанных на графе де Брюйна, оптимального с точки зрения предложенного метода оценивания качества геномной сборки. Исходный код доступен по ссылке: https://bitbucket.org/kromanenkov/apa.
3. На основе проведенного исследования существующих актуальных сборщиков генома и существующих методов объединения набора геномных сборок разработан и реализован распределенный алгоритм объединения результатов работ различных геномных сборщиков при отсутствии референса, основанный на кластеризации взвешенного графа контигов. В результате сравнения предложенного метода с существующими решениями на двух наборах данных было продемонстрировано, что разработанный метод превосходит их по большинству референс-зависимых метрик генома. Показана применимость предложенного метода для решения задачи объединения геномных сборок. Исходный код доступен по ссылке: https://bitbucket.org/kromanenkov/gar.

Предложенный метод оценки качества геномной сборки при отсутствии референса с помощью анализа частот k-меров может быть дополнен за счет взвешенного учета k-меров, соответствующих участкам повтора в геноме. Данная модификация позволит расширить область применимости метода на протяженные геномы более сложной структуры.

В рамках дальнейшего развития метода объединения результатов работ различных геномных сборщиков при отсутствии референса можно отметить следующие направления: усложнение алгоритма кластеризации для выделения областей графа, соответствующих контигам, полученных из одной хромосомы, и изучение возможности хранения контигов в сжатом виде для экономии оперативной памяти и возможности работы с большими геномами.

Благодарности. Автор выражает благодарность своему научному руководителю Якобовскому М.В. Автор благодарит Сальникова А.Н. за помощь в решении технических вопросов. Особую благодарность автор выражает сотрудникам отдела математических методов в биологии института имени А.Н. Белозерского: Алексеевскому А.В. и Спирину С.А. за ценные замечания и обсуждение результатов работы. Также автор благодарит Назипову Н.Н. и Тюльбашеву Г.Э. за помощь в интеграции программы в среду Mathcell. Автор благодарит свою семью и близких за поддержку: Романенкову Е.К., Романенкова В.А., Кулагину Г.Н., Гурченко Е.А.

Публикации автора по теме диссертации

1. *Романенков К. В., Сальников А. Н., Алексеевский А. В.* Параллельный метод объединения результатов работы программ по сборке генома // *Вестник ЮУрГУ, Серия Вычислительная математика и информатика*. — 2016. — Т. 5, № 1. — С. 24–34.

2. *Романенков К. В.* Метод оценки качества сборки генома на основе частот k-меров // *Препринты ИПМ им. М.В.Келдыша*. — 2017. — № 11. — С. 24.

3. *Romanenkov K. V.* Parallel merging method to integrate different genome assemblies // Proceedings of IEEE International Conference on Bioinformatics and Biomedicine (BIBM). — 2015. — Pp. 1461–1464.

4. *Романенков К. В., Сальников А. Н., Алексеевский А. В.* Параллельный метод объединения результатов работы программ по сборке генома // Параллельные вычислительные технологии (ПаВТ'2015): труды международной научной конференции. — 2015. — С. 456–462.

5. *Романенков К. В.* Методика оценивания качества геномных сборок на основе частот k-меров // Сборник тезисов XXIII Международной научной конференции студентов, аспирантов и молодых ученых «Ломоносов–2016», секция «Вычислительная математика и кибернетика». — 2016. — С. 62–64.

6. *Romanenkov K. V., Alexeevski A. V.* A new method of evaluating genome assemblies based on kmers frequencies // Proceedings of the 10-th Moscow conference on computational molecular biology. — 2017. — Pp. 1–3.

7. *Романенков К. В.* Объединение результатов работ программ по сборке генома. — URL: http://www.mathcell.ru/model8.php.

Романенков Кирилл Владимирович

Методы и алгоритмы автоматизированной сборки генома на вычислительном кластере

Автореф. дис. на соискание ученой степени канд. физ.-мат. наук

Подписано в печать ___.___._____. Заказ № _____
Формат 60×90/16. Усл. печ. л. 1. Тираж 100 экз.
Типография _____