# Разработка алгоритмов скаффолдинга при помощи дополнительной геномной информации

Клещин Антон Сергеевич, 16.Б10-мм

Научный руководитель: доц. каф. СП, к.т.н. Ю. В. Литвинов
Консультанты: доц. каф. стат. мод., к.ф.-м.н. А. И. Коробейников
Научный сотрудник Центра алгоритмической биотехнологии СПбГУ А. Д. Пржибельский
Рецензент: PhD кандидат Корнельского университета Д. А. Мелешко

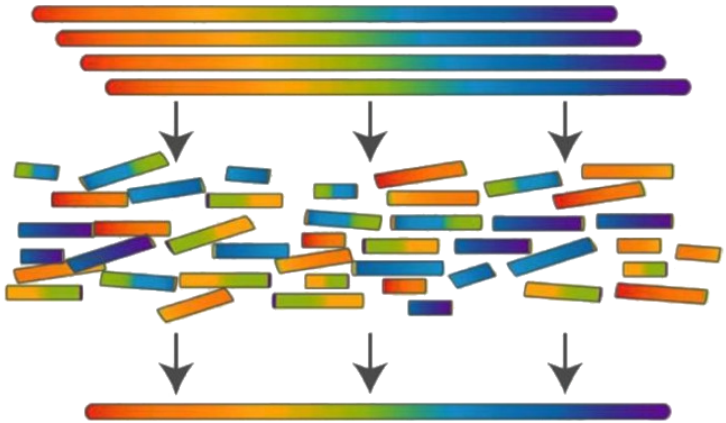
СПбГУ

9 июня 2020 г.



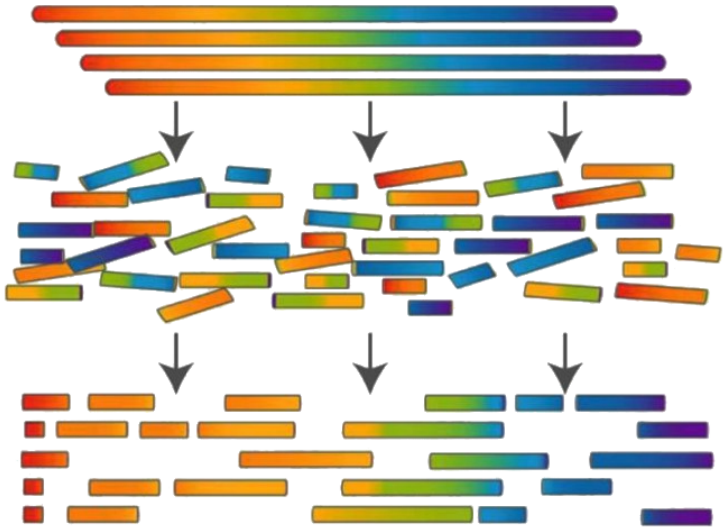
Добрый день, меня зовут Клещин Антон, и сейчас я расскажу о своей дипломной работе "Разработка алгоритмов скаффолдинга при помощи дополнительной геномной. информации"

# В идеальном мире



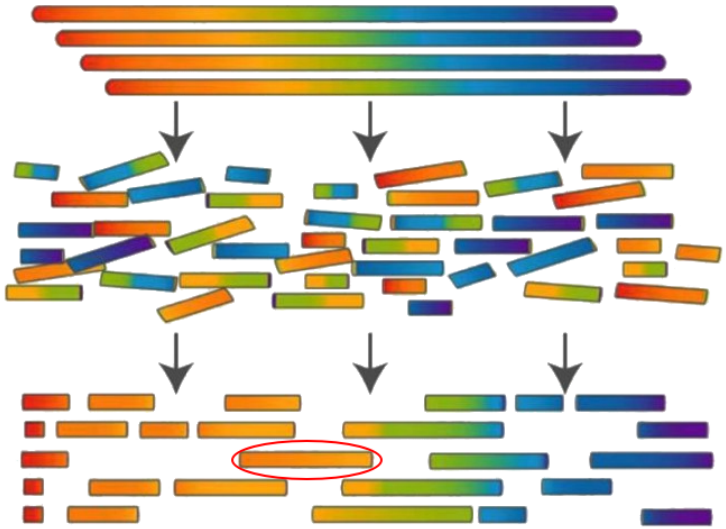
Несколько слов о том, что такое сборка генома.
Есть набор геномов, которые можно считать одинаковыми, и их пытаются прочитать и получают огромный набор кусочков, которые называются *ридами*. В идеале из них геномные сборщики могут восстановить всю ДНК.

# В реальности



Но на деле удаётся восстановить только некоторые фрагменты. Эти фрагменты называются *контигами*. Если известен порядок двух контигов и расстояние между ними, то это называется *скаффолд*. *Референсным геномом* называется эталонный геном для данного вида организма.

# В реальности



Основная идея работы состоит в том, что если кто-то скажет, что в геноме должна быть последовательность, похожая на ту, что обведена красным, то можно посмотреть на два контига выше, понять, что их края достаточно похожи на края выделенного фрагмента и затем объединить все три в один.

# Мотивация

- Переиспользование накопленных результатов
- Использование результатов сторонних ассемблеров
- Использование похожих геномов
- Упрощение метагеномной сборки использованием референсных геномов

Изначально всё затевалось для геномного ассемблера SPAdes под эгидой "переиспользования накопленных результатов", потому что есть исследователи, которым это нужно, а использование контигов под видом длинных ридов их не устраивало.

Также есть биологи, которые используют другой ассемблер для сборки из специального типа ридов, а затем используют его результат как дополнительную информацию в SPAdes.

Также есть два скорее теоретических применения:
Во-первых, это использование похожих геномов, таких как другие штаммы бактерий.
Во-вторых, это применение в метагеномике. В метагеномной сборке присутствуют риды не одного организма, а сразу множества. И если есть информация о том, какие организмы могут присутствовать в этом множестве, то можно использовать их референсные геномы для упрощения всей сборки.

# Постановка задачи

Цель — добавление поддержки контигов в качестве входных данных для геномного ассемблера SPAdes

- Разработка алгоритма скаффолдинга, использующего контиги
- Реализация расширения для геномного ассемблера SPAdes
- Тестирование алгоритма на известных геномах

Цель данной работы — добавление поддержки контигов в качестве входных данных для геномного ассемблера SPAdes. Для этого нужно было разработать алгоритм, использующий контиги, реализовать его и протестировать.

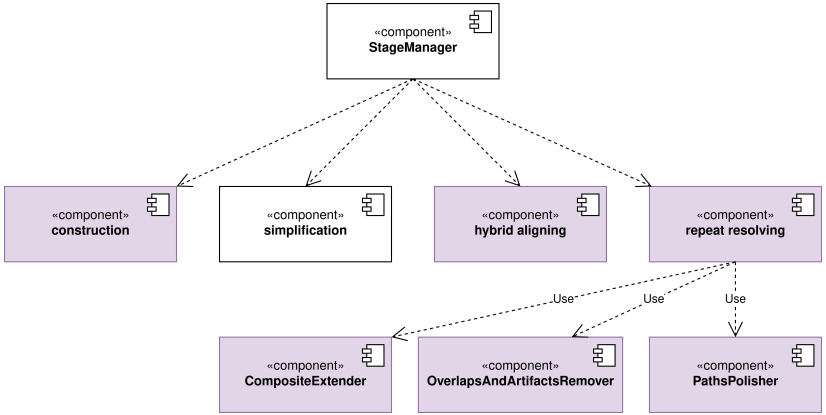
# Алгоритм

- Построение графа сборки
  - Короткие риды
  - Контиги
- Упрощение графа
- Выравнивание контигов на граф
- Выращивание путей
  - Точное совпадение с путями выравненных контигов
  - Неточное совпадение
- Постобработка

Сборка в SPAdes основывается на построении графа сборки — на его рёбрах находятся последовательности нуклеотидов, а путь в нём образует контиг. Поэтому основная идея алгоритма заключается в том, чтобы сначала выравнять входные контиги на граф сборки, то есть получить набор путей, образующих эти контиги (или их части), а затем использовать их как вспомогательные на этапе выращивания путей. На этом этапе берутся некоторые начальные пути графа и затем к их концам пытаются присоединять ребра.

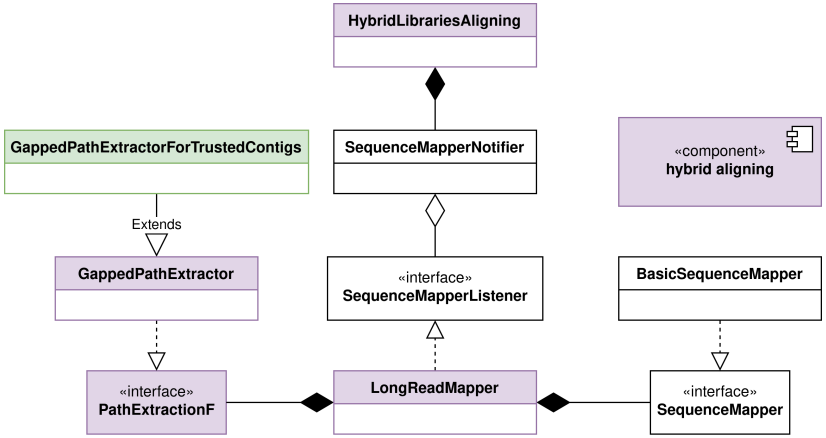
Но выравнивание на граф сборки — это очень вычислительнозатратная операция, поэтому происходит она не напрямую. Сначала контиги используются для построения графа сборки. Затем во время упрощения графа все рёбра, содержащие ошибки, удалятся, но при этом похожие рёбра объединятся. Теперь во время выравнивания контигов можно сравниваться не просто с подстроками рёбер и как-то их варьировать, а с уже заранее известными возможными вариациями. Благодаря этому выравнивание на граф происходит в сотни раз быстрее классических алгоритмов.

# Архитектура




На этом слайде изображен фрагмент архитектуры сборки в SPAdes. Здесь приведены компоненты, которые отвечают за этапы сборки, упомянутые в алгоритме. Фиолетовым выделены изменённые компоненты. Hybrid alining отвечает за выравнивание на граф, CompositeExtender занимается выращиванием путей, а OverlapsAndArtifactsRemover и PathPolisher выполняют постобработку путей. Остановимся подробнее на двух этапах.

# Архитектура

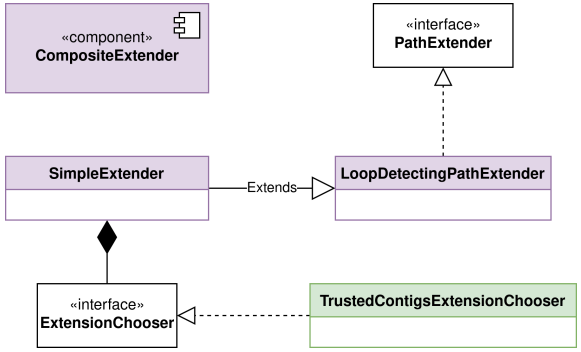



Первым будет выравнивание на граф. Зелёным выделен новый класс.

Этап выравнивания входных данных на граф сборки начинается с класса HybridLibrariesAligning. Он занимается подготовительной работой, создавая и конфигурируя классы, занимающиеся выравниванием соответствующих типов ридов, а затем связывает их с потоками входных данных через систему уведомлений SequenceMapperNotifier. После этого он начинает многопоточную обработку данных.

Для выравнивания контигов используется класс LongReadMapper, который параметризуется двумя алгоритмами: SequenceMapper и PathExtractionF. Первый занимается сопоставлением контига фрагментам рёбер графа. Эти фрагменты затем собираются, фильтруются, объединяются в рёбра, а затем и в пути вторым алгоритмом.

# Архитектура




На стадии выращивания путей создаётся набор PathExtender'ов в соответствии со входными данными. Каждый PathExtender позволяет продолжить переданный ему путь на одно ребро. С помощью CompositeExtender все созданные PathExtender'ы применяются последовательно в соответствии с некоторым приоритетом, пока одному из них не удастся продолжить путь.

Одной из реализаций PathExtender является SimpleExtender. Он параметризуется алгоритмом ExtensionChooser, который по пути определяет, какие рёбра возможны для его продолжения. В случае если продолжение единственно, методами LoopDetectingPathExtender проверяется, можно ли добавить его.

# Метрики

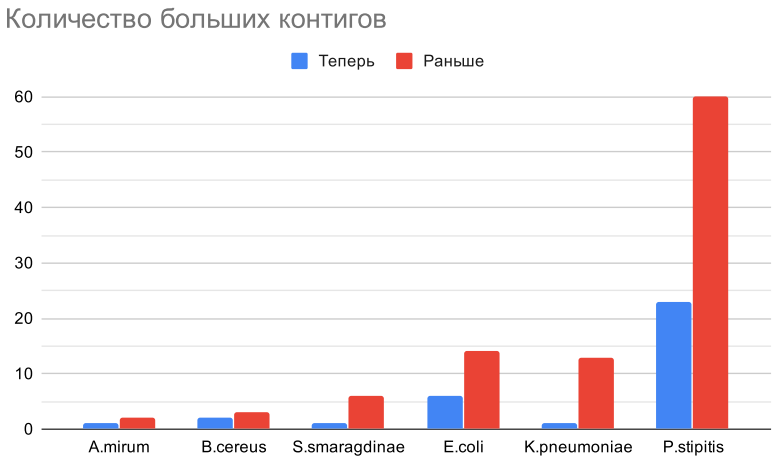
- Количество больших контигов
- Количество структурных ошибок

В данной работе были использованы четыре метрики, но для демонстрации результатов достаточно двух из них. Это количество больших контигов и количество структурных ошибок.

Уменьшение числа больших контигов означает, что удалось объединить несколько других больших контигов в один. Поэтому чем их меньше, тем лучше.

Структурные ошибки — это места в контигах, в которых соединения быть не должно. То есть соединения есть в контиге, но отсутствуют в референсном геноме.

# Тестирование

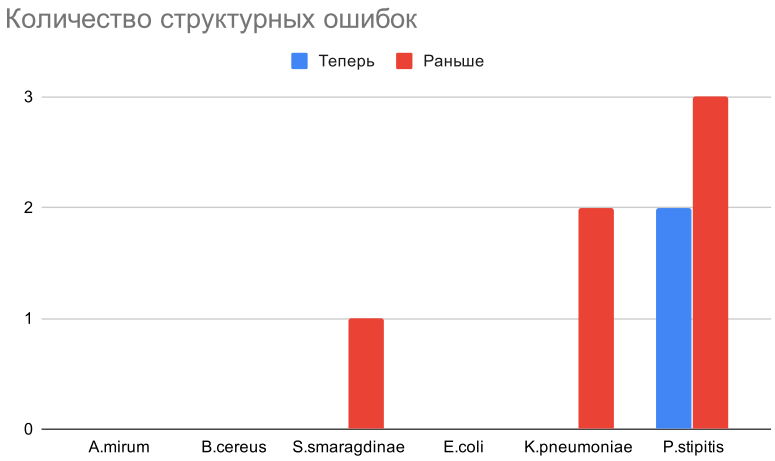



В SPAdes уже существовала возможность использования контигов в качестве входных данных, но она фактически являлась набором алгоритмов для длинных ридов, а не контигов. Так как контиги можно рассматривать как длинные риды, то этот подход работал, но, очевидно, далеко не самым оптимальным образом.

На этом слайде представлены результаты некоторых сборок одиночных организмов. В качестве входных контигов использовались референсные геномы.

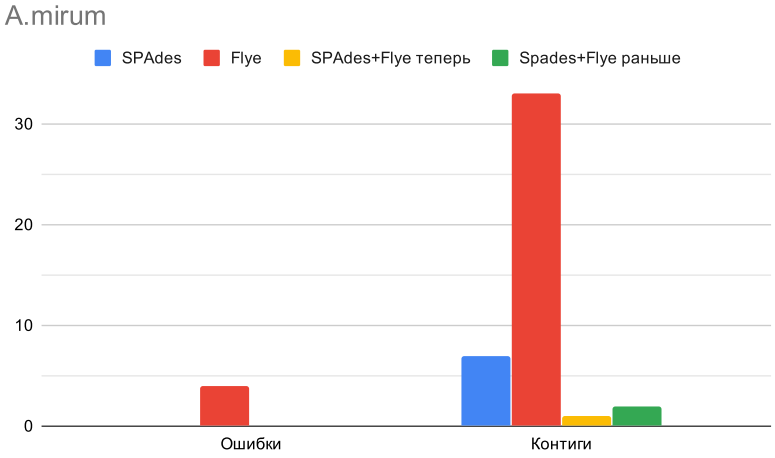
Первые пять элементов являются бактериями, а последний — грибом с девятью хромосомами (то есть минимум девять контигов должно быть). У грибов геном значительно сложнее, поэтому граф сборки получается запутаннее, и поэтому такое резкое ухудшение результатов по сравнению с остальными геномами вполне ожидаемо.

# Тестирование




Из этого графика видно, что количество структурных ошибок также уменьшилось.

# Тестирование, Flye

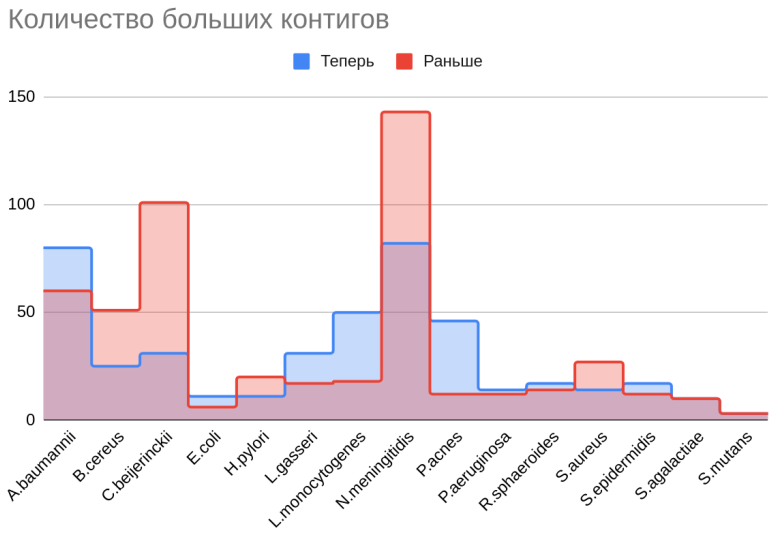



В начале презентации я упоминал биологов, которые используют сторонний ассемблер. Они используют Flye для сборки из специального типа ридов, получают не очень хороший результат сборки, который дальше используют в SPAdes уже со всеми доступными им ридами.

Из графиков видно, что добавление результатов Flye дало прирост по сравнению с чистым Spades.

Здесь результаты не говорят о плохой работе Flye, ему просто дают такие входные данные, что лучше сделать сложно.

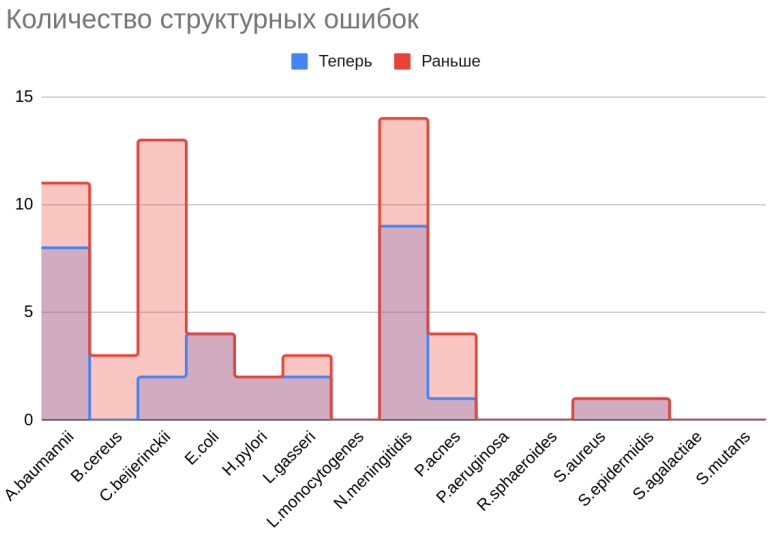
# Тестирование, метагеном




На этом слайде показаны результаты тестирования на метагеномной сборке. Риды всех бактерий с графика передавались в ассемблер одной кучей. В качестве контигов передавались референсные геномы бактерий.

Видно, что для некоторых бактерий количество больших контигов возросло, то есть результаты по этой метрике ухудшились.

# Тестирование, метагеном




Но здесь главным достижением является то, что удалось существенно уменьшить количество структурных ошибок. Это важнее, потому что биологу найти ошибку в сборке значительно сложнее, чем объединить контиги.

# Заключение

- Разработан алгоритм скаффолдинга, использующего контиги
  - Алгоритм выравнивает контиги на граф сборки, а затем использует полученные пути при разрешении повторов
- Реализовано расширение для геномного ассемблера SPAdes
  - Реализовано на языке C++
  - Расширение позволяет эффективно использовать контиги в качестве входных данных
  - Исходный код SPAdes доступен по ссылке: `https://github.com/ablab/spades/`.
- Алгоритм протестирован на известных геномах
  - Протестировано на сборках одиночных геномов с высоким и низким качеством входных контигов, а также на метагеномной сборке
  - Теперь соединяется больше контигов с меньшим количеством ошибок по сравнению с предыдущим модулем SPAdes



Итак, в ходе данной работы были достигнуты следующие результаты. Разработан алгоритм скаффолдинга, использующий контиги. Реализовано расширение для геномного ассемблера SPAdes на языке C++. Оно войдёт в состав следующего релиза. А также проведено тестирование на известных геномах, и оно показало, что теперь соединяется больше контигов с меньшим количеством ошибок по сравнению с предыдущим модулем SPAdes.