ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
Факультет Санкт-Петербургская школа физико-математических и компьютерных наук

Кунявская Ольга Александровна

СТРУКТУРНЫЙ И ЭВОЛЮЦИОННЫЙ АНАЛИЗ ЧЕЛОВЕЧЕСКИХ ЦЕНТРОМЕР

Выпускная квалификационная работа (МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)
по направлению подготовки 01.04.02 Прикладная математика и информатика
образовательная программа «Программирование и анализ данных»

Рецензент
к.т.н.
_____
А.А. Сергушичев

Научный руководитель
д.ф.-м. н., проф.
_____
А.В. Омельченко

Консультант
к. ф.-м. н., проф.
_____
П.А. Певзнер

Санкт-Петербург
2021

Оглавление

Термины

Мономерный блок – один фрагмент длиной примерно в 171 нуклеотид на который разбивается центромера. Мономерные блоки похожи друг на друга на 65-100%.

Мономер – кластер мономерных блоков. Внутри кластера мономеры похожи на 95-100%. Иногда под терменом мономер буду иметь введу не сам кластер, а его центр(усредненный мономерный блок)

Мономерный консенсус – центр кластра мономерных блоков(усредненный мономерный блок).

Гибридный мономер – мономер, который можно представить как конкатинацию префикса и суффикса двух других мономеров.

Моноцентромера – центромера записанная в алфавите мономеров

Канонический HOR – Строка из мономеров из которой путем копирования произошла остальная центромера. Строка специфична для центромеры, HOR расшифровывается, как high-order repeat.

Вариация HOR – строка, которая была получена из канонического HOR путем делеций

HOR – high-order repeat, канонический HOR, либо его вариация. Строка.

HOR фрагмент – один повторяющийся участок в моноцентромере, который является представителем либо канонического HOR, либо его вариации. Конкретное вхождение строки в моноцентромеру.

CE постулат – постулат центромерной эволюции, который гласит, что существует канонический HOR из которого произошла вся центромера. При этом в канонический HOR входят все частые не гибридные мономеры ровно по одному разу.

Расхождение –

$$div(S1, S2) = 100 \cdot \frac{edit\_distance(S1, S2)}{max(|S1|, |S2|)}$$

$edit\_distance$ – редакционное расстояние

Похожесть – говорим две строки $S1$, $S2$ похожи на X%, где

$$X = 100 - div(S1, S2)$$

Сепарации – *Сепарацией* мономера M(separation(M)) минимальное расстояние между M и другими мономерами.

Радиус – *Радиус* мономера M(radius(M)) наибольшее расстояние между мономерным консенсусом и его мономерными блоками.

Коэффицент сепарации –

$$separationRation(M) = \frac{separation(M)}{radius(M)}$$

Введение

Актуальность изучения центромер.

Центромерные сателитные повторы – это наиболее длинные и трудные для сборки тандемные повторы в человеческом геноме и поэтому задача сборки человеческих центромер не была решена до недавнего времени. Поэтому все предыдущие исследования о связи геномной последовательности и заболеваний игнорировали примерно 3% человеческого генома. При этом центромеры играют ключевую роль в процессе деления клетки и многие генетические заболевания связаны с нарушениями в количестве хромосом, которые появляются в процессе мейоза([17]). Более того, вариативность в центромерах связывают с возникновением рака и бесплодия([21], [14], [15]). Так же изучение центромерных последовательностей интересно с точки зрения решения открытых проблем связанных с эволюцией центромер([19], [20], [24], [22]) и центромерным парадоксом([13]), который заключается в том, что центромеры выполняют очень консервативную функцию и при это очень быстро эволюционируют.

Мономеры.

Центромеру можно разбить на фрагменты длиной примерно 171 нуклеотид, так, что бы фрагменты были похожи между собой хотя бы на 65%. Эти фрагменты называются *мономерный блоки*. То есть, центромера представляет из себя тандемный повтор. Далее мономерные блоки можно разбить на кластера так, что бы внутри кластера мономерные блоки были похожи друг на друга на 95-100%. Эти кластера будем называть *мономерами* и теперь каждый мономерный блок можно отнести к одному из мономеров. То есть теперь, центромеру можно записать над алфавитом мономеров и получить *моноцентромеру*. Для каждого кластера можно выделить центр(усредненый мономерный блок), его я буду называть *мономерным консенсусом*. Одна центромера обычно состоит

из порядка 20 000 мономерных блоков. При этом количество повторов может варьироваться внутри человеческой популяции([5]).

Далее я могу упоминать слово *мономер* либо подразумевая кластер мономерных блоков, либо центр этого кластера(мономерный консенсус).

Постулат центромерной эволюции.

Согласно современной теории можно сформулировать *постулат центромерной эволюции*(*CE постулат*). Постулат гласит, что существует некоторая предковая строка из мономеров из которой путем копирования и делеций произошла вся центромера. Такую строку называют *канонический HOR*(high-order repeat). Эволюционно в процессе делеций могли образовываться структурные вариации канонического HOR. Например, пусть канонический HOR состоит из мономеров ABCDEF, тогда в процессе эволюции из-за делеций могла образоваться вариация HOR ABF. Более того, возможно образование *гибридного мономера*([7], [6], [11]), когда в процессе делеции был вырезан суффикс одного мономера и префикс другого, то есть возможна вариация AB(C/E)F. Будем называть HORом либо канонический HOR либо его вариацию.

CE постулат можно описать следующим образом:

- Каждая современная центромера была получена из единственного канонического HOR, который был сформирован из нескольких разных предковых мономеров.

- Каждый частый не гибридный мономер в центромере был получен из единственного предкового мономера. Количество предковых мономеров равно количеству частых не гибридных мономеров в данной центромере.

- Каждый гибридный мономер был получен путем конкатинации пары(а иногда и большего количества) предковых мономеров.

- Кроме фрагментов полученных из канонического HORa, существуют фрагменты, которые образованы частью канонического

HORa(то есть какой-то подстракой канонического). Все другие фрагменты называются *вариациями* HORa. И хотя для большенства человеческих центромер, самый частый HOR является каноническим, это далеко не всегда так.

Структура центромеры

Таким образом, структуру центромеры можно представить как вложенный тандемный повтор, где центромеру можно представить из повторяющихся HOR фрагментов. А каждый HOR в свою очередь состоит из повторяющихся мономерных блоков. Центромера может содержать порядка нескольких 1000 HOR фрагментов, а каждый HOR фрагмент содержит порядка 10 мономеров.

Для примера рассмотрим центромеру X. Структура центромеры X показана на рисунке 1. Большая часть HOR в центромере X состоит из 12 мономеров. Хотя разные HOR фрагменты в центромерер X очень похожи(сходство 95-100%), 12 мономеров образующих HOR достаточно разные(сходство 65-88%). Более того, кроме стандартных 12мономерных HOR фрагментов, структура некоторых HOR в центромере X не каноническая: 35 из 1510 фрагментов образовано меньшим или большим количеством мономеров, чем канонический 12-мерный HOR([6]).

Задача аннотации центромер.

Недавний прорыв в секвенирование длинных ридов и алгоритмов в биоинформатике в последние годы позволили собрать человеческие центромеры([6], [16], [18]) и впервые появилась возможность начать изучение структуры и эволюции человеческих центромер. В недавних работах, связанных с эволюцией центромер([7], [6], [22]), была показана важность декомпозиции центромеры на мономеры и HORы.

Задачу об аннотации можно сформулировать следующим образом. Изначально дана центромерная последовательность: строка над алфавитом AGCT. На выходе хотим получить её представление в виде моноцентромеры и декомпозиции на канонический HOR и его вариации. В

этой задаче можно выделить следующие этапы: (1)декомпозиция на мономерные блоки, (2)выделение мономеров, (3)представление в виде моноцентромеры, (4)поиск канонического HOR, (5)декомпозиция на HOR.

Полуручная аннотация.

На текущий момент существует одна сборка человеческих центромер и практически закончена их аннотация полуручными методами ([7]). При этом в ближайшее время планируется полная сборка 100 человеческих геномов и в том числе планируется аннотация центром в этих геномах и сравнение этих геномов между собой. При этом центромеры – это очень вариабельные регионы и решение для одного человека может не подходить для другого. Кроме человека, планируются сборки разных видов обезьян, свиней и других видов животных. Из-за этого появляется необходимость автоматизировать аннотацию центромер.

Инструменты.

Существует ряд инструментов связанных с некоторыми этапами задачи аннотации. Поскольку сборка появилась только в 2019 году большая часть инструментов рассчитана на другие данные и решает немного другую задачу. Сейчас я подробнее расскажу про каждый из инструментов.

colorHOR([8]): инструмент появился в 2004 году и его цель визуализация центромеры для получения представления о её структуре. То есть этот инструмент может служить чем-то вспомогательным для аннотации, но ни в коем случае не самой автоматической аннотацией. В некотором смысле можно считать что инструмент в итоге получает раскраску центромерам по мономерам, но в очень творческом смысле.

Для разбиения мономеры на блоки в этом инструменте берется некоторая строка из 6 нуклетотидов и утверждается, что она содержится в любом мономере. В целом, поскольку мономерные

блоки похожи между собой на 65% утверждение не супер безумное. Дальше для разбиения центромеры на блоки, центромера разбивается по этой строке из 6 нуклеотидов. В результате, в этой работе разброс мономеров от 60 до 400 нуклеотидов. Это очень неправдаподобный результат, поскольку обычно мономеры имеют длину в 171 нуклеотид и адекватный разброс это от 160 до 190. Поэтому кажется – это не самый лучший способ разбиения на мономерные блоки.

Следующий момент это кластризация мономеров. В этом инструменте два мономерных блока относятся к одному мономеру если они имеют одинаковую длину. То есть кластеризация происходит по длине мономера. То есть данная кластеризация не учитывает возможность вставок/удаленний в мономеры и то, что несколько разных мономеров могут иметь одинаковую длину.

Следующие этапы декомпозиции не производятся.

HORdetect([19]) Инструмент появился в 2007 году и на тот момент сборок центромер не было и даже не было длинных ридов. Но при этом было достаточно популярное секвенирование второго поколения – то есть короткие риды. Этот инструмент из коротких ридов пытается восстановить множество HOR. В данном случае сложность дополнительно заключается в том, что в одном риде может быть всего несколько мономеров, но не HOR целиком.

Первом шагом производится разбиение на мономерные блоки с помощью RepeatMasker. Этот инструмент ищет повторы, но при этом не специализирован непосредственно на поиске мономерных блоков. От этого при его работе могут возникать ошибке, подробнее про это в статье [11]. Далее выделенные мономерные блоки кластиризуются. в ней по очереди рассматриваются мономерные блоки. Сначала берется первый мономерный блок и из него создается отдельный кластер, дальше берется следующий мономерный блок и перебираются все существующие кластера, если до какого-то из центров кластеров расстояние меньше заданного па-

раметра, то тогда мономерный блок относится к этому кластеру и консенсус в нем пересчитывается, иначе он образует новый кластер. Описанная кластеризация очень простая и я даже думаю, что в большинстве случаев будет неплохо работать. Меня однако в ней очень смутило, что кластеризация будет зависеть от того, в каком порядке я мономерные блоки рассматриваю. То есть я могу придумать искусственный пример, в котором в одном порядке образуется одно количество кластеров, а если удалить первый мономерный блок, то количество кластеров будет в два раза большим.

Прежде всего меня смущает именно такая потенциальная нестабильность, поскольку сложно точно определить точные координаты центромеры и хочется, что бы если при выделение центромеры я взяла на пару мономеров меньше кластеризация оставалась бы примерно такой же.

Далее для предсказания HOR брались все возможные тройки мономеров. И если есть тройка (a, b, c) и тройка (b, c, d), в которой последние два мономера совпадают с первыми двумя в следующей тройке, то тогда эти триплеты объединяются в четверку (a, b, c, d). Это решение хорошее при учете, что работа происходит с короткими ридами. Однако проблема в выделение мономеров сразу приведет к неправильным HOR, а поправки множества мономеров никакой не происходит. И более того, из-за того что рассматривается столь ограниченная информация, то возможно, что в множестве будут содержаться несуществующие HOR. У меня же в моем решение есть возможность использовать больше контекста, для более точного решения.

Alpha-CENTAURI([2]) 2016 год и в этом инструменте уже работают с длинными ридами. Для выделения мономерных блоков используется предобученная hmm(hidden markov model), которая уже зашита в тул. Тут проблема в отсутствие гибкости. Если я например захочу обобщить инструмент на другие виды животных, то тогда разбиение на блоки перестанет работать и станет непонятно,

каким образом можно будет обобщить решение.

Далее мономерные графы кластризуются. В статье не сказано, как они кластеризуют, но если посмотреть код, то можно выяснить метод. Они строят граф, вершинами которого являются мономерные блоки, а ребро проводится, если мономерные блоки находятся на небольшом расстояние друг от друга. Далее кластер – это компонента связности.

Проблема в этом способе кластеризации заключается в том, что при таком подходе может случайно образовываться кластер с очень большим радиусом и таким образом много разных мономеров будет объединяться в один кластер. То есть если мы хотим решить задачу, в которой все мономеры находятся на небольшом расстояние от центра кластера, то таким образом мы не можем её решить.

Далее непосредсвенно выделение HOR не происходит. Дальше просто все риды делятся на те, в котором все HOR одинаковые и в котором разные. При этом понятие HOR не определяется и рассматривается только расстояние между мономерами и если расстояния всегда одинаковые. то утверждается, что HOR состоит из одних и тех же HOR, если расстояние разные, то в риде разные HOR. При этом, в случае возникновения проблем с HOR будет возникать проблема и с детекцией видов. Канонический HOR тут не выводится и не производится дополнительная декомпозиция.

StringDecomposer([11]) Для заданной нуклеотидной строки $Centromere$ и множество мономеров $Monomers$, $StringDecomposer$ декомпозирует $Centromere$ на мономерные блоки(каждый блок похож на один из мономеров) и переводит строку $Centromere$ в моноцентромерную строку $Centromere^*$ над алфавитом мономеров. Для каждого мономера M $StringDecomposer$ выдает множество $M-$блоков из центромеры(мономеры, которые более близки к мономеру M, чем к другим). Однако, задача выделения множества мономеров

оказалось за рамками рассмотрения в *StringDecomposer*.

Проблемы

Не определено понятия HOR и мономеров. Самая глобальная проблема заключается в том, что нет конструктивного определения ни понятия HOR, ни понятия мономеров. То есть нет ни целевой функции, которую можно было бы оптимизировать для вывода HOR(мономеров), ни какого-либо способа проверки качества двух разных решений. Канонический HOR определяется как предковая строка мономеров и не уточняется ни каким образом её можно вывести, ни как проверить, что эта строка является верной и единственной.

Хотя CE постулат является общепринятым, мне не известно ни доказательство этого постулата ни алгоритма, который бы для данной центромеры выводил бы канонический HOR. Более того, концепция HOR зависит от параметров, и для одних параметров CE постулат может работать, а для других уже нет. При этом остается не ясным каким образом выбирать параметры такие как пороговое значение частоты(в каком случае мономер следует считать частым), пороговое значение похожести(при какой степени схожести следует считать, что данный мономерный блок относится к данному мономеру), пороговое значение для гибридов(в каком случае стоит считать, что мономер является гибридов двух других). Например, HOR для центромеры X состоит из 12 мономеров, этот 12 мономерный HOR произошёл из 5мономерного HORa([3]). До конца не ясно, алгоритм выведения канонического HORa должен выводить 12мономерный HOR или 5мономерный HOR для центромеры X.

Существующие решения не согласованы с CE постулатом. Во всех решениях выделение мономеров никак не согласовывается с концепцией HOR, а концепция HOR очень зависит от параметров, которые использовались при выводе мономеров. Во

всех решениях выделение мономеров и HOR рассматриваются как две независимые задачи и полученные мономеры и HOR никак не валидируются. И действительно, нет целевой функции для оценки качества результата. Лучшее, что мы сейчас имеем это CE постулат, то есть некоторый набор свойств, который должен выполняться для набора множества мономеров и HOR. И для того, что бы решение получалось согласованным с текущей теорией необходимо для каждого этапа аннотации учитывать возможные ошибки, которые могли появляться на предыдущих этапах и уточнять решения предыдущих этапов имея новую информацию.

Текущие решения не рассматривают гибридов Ещё один пробел, ни в одной из предыдущих работ для автоматической аннотации не рассматривались гибридные мономеры. В работах [7], [6], [11] было показано, что часто в центромерах возникают *гибридные* мономеры(мономеры полученные в результате конкатенации половинок других мономеров), выведением которых мало занимались. Есть гипотеза, что новые мономеры возникают именно из гибридов.

HOR не всегда описывает архитектуру. Понятия канонического HOR скорее направлено на описание предковой мономерной строки из которой произошла текущая центромера. При этом только выделения канонического HOR может плохо отражать непосредственно структуру центромеры. То есть нет компактного способа описания вариаций в центромере и её строения.

Краткое описание проекта.

В рамках данной магистерской работы я разрабатывала модули, которые являются частью более большого проекта *CentromereArchitect*. В проекте *CentromereArchitect* мы ставили своей целью разработать инструмент для автоматической аннотации центромер от начала и до

конца. В него входит выделение мономеров(MonomerGenerator), декомпозиция центромеры на мономеры(StringDecomposer), выделение канонических мономеров и HOR(HORmon) и декомпозиция центромеры на HORы(HORdecomposer). Я занималась только задачами о выделение мономеров и о выделение канонических HORов и мономеров, а задачами декомпозиции центромеры на мономеры и HORы занималась моя коллега Татьяна Дворкина. Все стадии аннотации центромеры в рамках проекта *CentromereArchitect* показаны на рисунке 2.

Кроме того, в рамках этой работы предложена альтернативная концепция представления архитектуры центромер, которая более полно отражает информацию о строение центромеры, чем описание центромеры в виде канонического HORa.

Цель и задачи.

Цель: разработать вычислительный метод для автоматического выделения мономеров и канонических HOR для человеческих центромер.

Задачи:

- разработать и реализовать алгоритм для автоматического выделения мономеров
- выделить мономеры и сравнить полученные мономеры с известными результатами из более ранних работ
- разработать алгоритм для выделения канонических HORов
- разработать модель для описания структуры центромеры

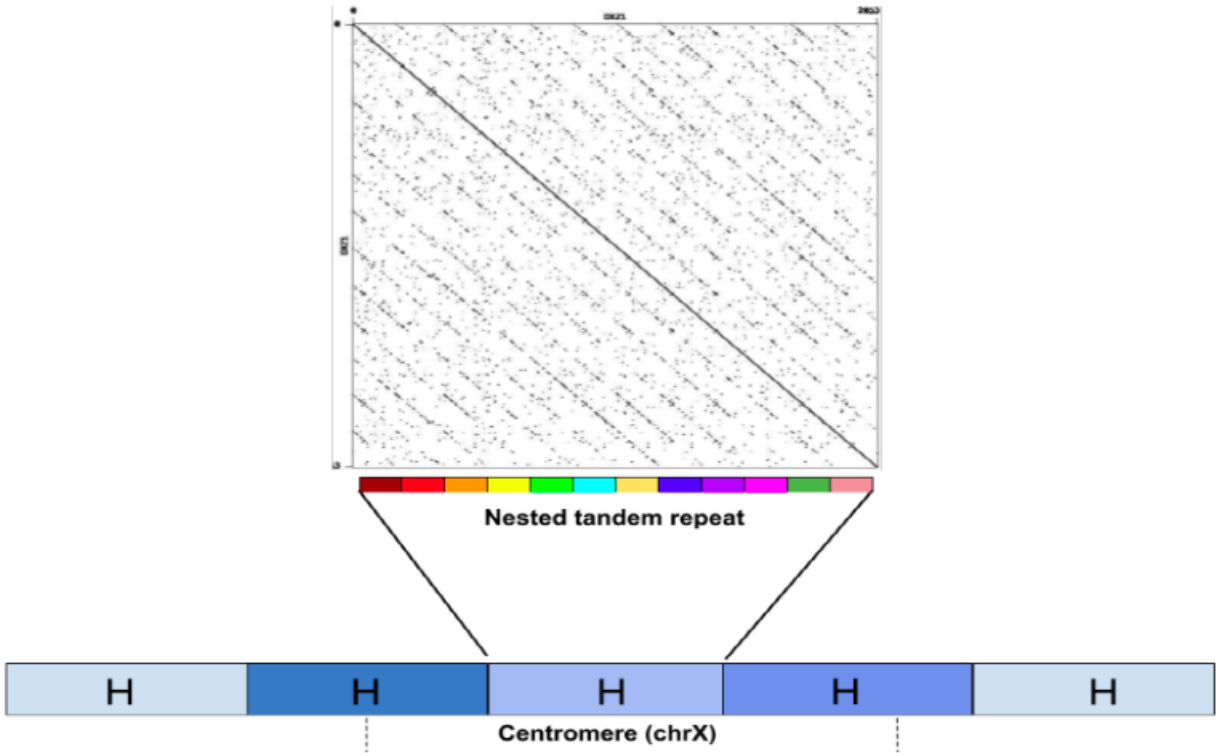


Рис. 1: **Архитектура центромеры в хромосоме X** Надвно собранная центромера хромосомы X состоит из 18103 мономеров длины примерно 171bp каждый и единственного LINE элемента согласно сборки [6](в последней T2T сборке([18]) изменения небольшие). Эти мономеры организованы в 1510 повторов большого порядка(high-order repeats, HORs). 5 HORs на рисунке покрашены в 5 оттенков синего, иллюстрируя вариации HORов. Каждый HOR состоит из вложенного тандемного повтора состоящего из мономеров. Большинство HORов в центромере X – это *канонический* HOR фрагмент, который образован 12 мономерами(на рисунке изображен 12 разными цветами). Картинка наверху соответствуют точечному графику канонического HOR, демонстрирующего сходство разных мономеров между собой. HORы между собой похожи на 95-100%, а мономеры на 65-88% между собой. Кроме 12-мономерного HORa, есть еще небольшое количество не канонических HORов с разным числом мономеров.

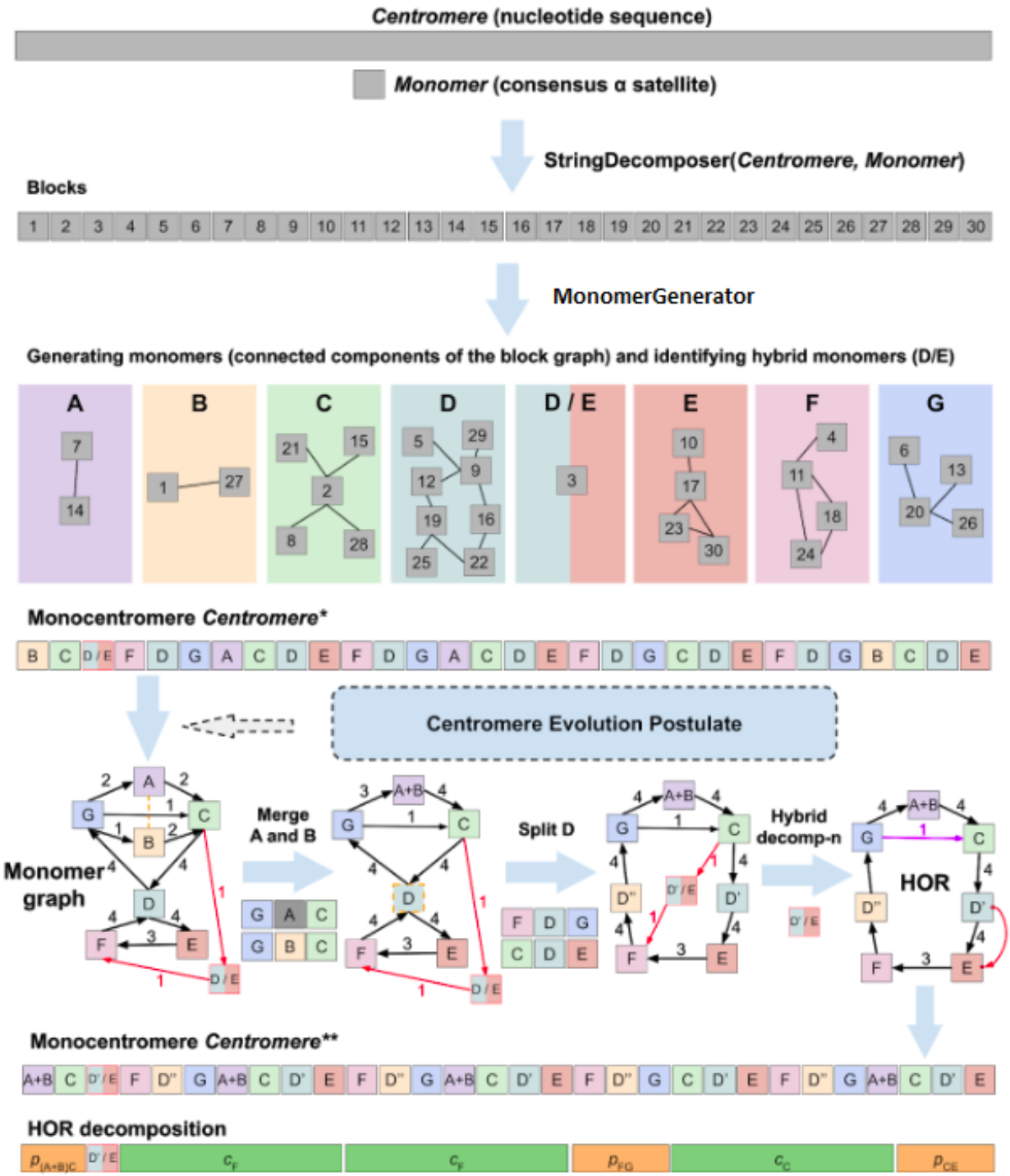


Рис. 2: **Этапы CentromereArchitect** Для заданной нуклеотидной последовательности *Centromere* и консенсусной альфа-сателитной последовательности *Monomer*, *HORmon* запускает *StringDecomposer* для разделения центромеры на мономерные блоки. После этого запускается *MonomerGenerator* что бы кластеризовать похожие мономерные блоки на мономеры и находит гибриды(обозначение D/E обозначает гибрид мономера D и E) и превращает центромеру в моноцентромеру *Centromere**. После этого, *HORmon* использует сгенерированную моноцентромеру для построения мономерного графа(красные ребра сединяют гибрид с остальным мономерным графом). Для того, что бы соответствовать CE постулату, HORmon разделяет и склеивает мономеры и декомпозирует гибрид. Оранжевое пунктирное ребро соединяет мономеры A и B показывая что они являются кандидатами для слияния. Разделяемый мономер D показан пунктирным оранжевым, что бы показать, что мономер разделяемый. Гибридная декомпозиция заменяет гибридную вершину D'/E одним «красным» ребром, которое соединяет префикс D' с суффиксом E. Операции разделения, слияния и гибридной декомпозиции приводят к новому набору мономеров и преобразуют *Centromere* в моноцентромеру *Centromere***. Черный цикл на графике мономеров *Centromere*** представляет *HOR*; пурпурный край, соединяющий мономеры G и C, представляет собой низко-покрытую хорду в этом цикле. HORdecomposer использует этот *HOR* для генерации *HOR*-разложения *Centromere*** на канонические (cF, cC), частичные (p (A + B) C, pFG, pCE) и вспомогательные (одиночный блок D '/ E) HOR.

1. Выделение мономеров

1.1. Данные

Центромеры

Для извлечения мономеров я использовала сборку(public release v1.0) CHM13 гаплоидной линии клеток(https://github.com/nanopore-wgs-consortium/chm13#v10), которая была получена в рамках Telomere-to-Telomere (T2T) консорциума ([16], [18], [25])

В таблице 1 представлены координаты живых регионов человеческих центромер для сборки CHM13. Эти координаты были получены в ручную моим коллегой и для дальнейшей работы по извлечению мономеров использовались именно эти регионы.

Chromosome	**start**	**end**
1	121 796 218	126 300 656
2	92 333 539	94 673 018
3	91 735 618	92 596 313
3	92 869 954	92 903 597
3	95 863 962	96 415 434
4	49 705 249	50 433 651
4	52 115 581	54 870 604
4	54 980 385	55 199 889
5	47 039 130	47 049 658
5	47 077 199	49 596 619
6	58 286 939	61 058 622
7	60 414 370	63 714 496
8	44 243 543	46 325 076
9	44 952 791	47 582 587
10	39 633 786	41 881 061
11	51 035 791	54 413 485
12	34 620 831	37 202 143
13	16 220 361	18 181 363
14	10 149 798	12 766 096
15	17 263 917	18 279 251
16	35 848 293	37 829 526
17	23 433 664	27 571 610
18	15 965 700	20 933 550
19	25 821 756	29 768 168
20	26 925 847	29 099 648
21	11 699 868	12 043 391
22	12 816 950	15 739 833
X	57 817 899	60 927 196

Таблица 1: **Координаты живых регионов человеческих центромер.** Хромосомы 3(4, 5) содержат три(три, два) участка альфа-сателитных регионов, которые разделены немономерными регионами длинною 274kb и 2960kb(1682 kb и 110kb; 27kb)

Консенсусный мономер

Далее для корректной работы алгоритма который будет описан ниже мне потребуется консенсусный мономер. Для этого я использовала мономер типа A(cons_A_type_AJ131207|modified|12-àG|44 deleted) – консенсус извлеченный из человеческого генома([4]). Неизвестные нуклеотиды "N"в этой последовательности были заполнены в соответствии с большинством голосов в матрице профиля нуклеотидов([4]), что привело к следующей последовательности:

AATCTGCAAGTGGACATTTGGAGCGCTTTGAGGCCTA
TGGTGGAAAAGGAAATATCTTCACATAAAAACTAGAC
AGAAGCATTCTCAGAAACTTCTTTGTGATGTGTGCAT
TCAACTCACAGAGTTGAACCTTTCTTTTGATAGAGCA
GTTTTGAAACACTCTTTTTGTAG

1.2. Задача выделения мономеров

Для данный строки $Centromere$ и заданного множества строк $Monomers$ инструмент $StringDecomposer$([11]) декомпозирует $Centromere$ на (мономерные) блоки(будем обозначать итоговое множество мономерных блоков как $Blocks(Centromere, Monomers)$). Для каждого блока $Block$, $StringDecomposer$ сопоставляет значение $div_1(Block)(div_2(Block))$, что соответствует расхождению между мономерным блоком и наиболее близким мономером(вторым по близости). Расхождение между двумя строками определяется как редакционное расстояние деленное на максимальную длину двух строк.

Для данного мономера M из мономерного множество $Monomers$ я обозначаю блок из $Blocks(Centromere, Monomers)$ как $M-$блок если M наиболее близкий мономер к этому блоку. Определим $M-$консенсус как консенсус множественного выравнивания всех $M-$блоков. Для двух мономеров M и M' обозначим редакционное расстояние между M-консенсусом и M'-консенсусом как $distance(M, M')$. *Сепарацией* мономера M(обозначим как $separation(M)$) определим как наименьшее расстояние между M и всеми другими мономерами. *Радиус* мономера M(обозначим как $radius(M)$) определим как наибольшее редакционное расстояние между M-консенсусом и всеми M-блоками. Определим *коэффицент сепарации* мономера M как $separationRation(M) = \frac{separation(M)}{radiusM}$. Мономер M хорошо сепарирован, если для любого мономера M' выполняется $distance(M, M') > radius(M) + radius(M')$.

Количество мономера M в центромере(обозначим $count(Centromere, M)$) определяется как количество M-блоков в $Blocks(Centromere, Monomers)$. Мономер определяется как частый, если его количество превышает $\frac{|Blocks(Centromere,Monomers)|}{FreqCeiing}$(значение по умолчанию для $FreqCeiling = 40$). В противном случае мономер определяется как нечастый. Нечастый мономер определяется как редкий, если его количество не превосходит значения $rareMonomerCount$(значение по умолчанию $rareMonomerCount = 3$).

Скажем, что блок $Block$ является *разрешённым*, если $div_1(Block)$

меньше порога $maxResolvedDivergence$(значение по умолчанию $maxResolvedDivargence = 5\%$). Мы обозначаем блок $Block$ как не мономерный, если $div_1(Block)$ превышает пороговое значение $maxDivergence$(значение по умолчанию $maxDivergence = 40\%$). И $Block$ является *неразрешённым* если он не является ни *разрешённым*, ни не мономерным.

Будем говорить, что мономерное множество $Monomers$ *разрешает* центромеру $Centromere$ если доля разрешенных блоков превосходит некоторое пороговое значение $FractionResolvedBlocks$ и все остальные блоки являются не мономерными(значение по умолчанию $FractionResolvedBlocks = 0.95$). Пусть задан целочисленные параметр $Length$, назовём множество мономеров $Length - uniform$ если у всех мономеров длина близка к $Length$, то есть разница длин не превышает параметра $MaxLengthDivergence$(значение по умолчанию $0.03 \cdot Length$)

Задача выделения мономеров

Вход: Строка $Centromere$ и параметры $maxResolvedDivergence$, $Length$, $MaxLengthDivergence$ и $FractionResolvedBlocks$

Выход: $Length - uniform$ множество мономеров $Monomers$, которые разрешают центромеру $Centromere$ и имеют минимальное количество мономеров среди всех $Length - uniform$ множества мономеров, которые разрешают $Centromere$.

В предыдущей попытке сгенерировать всё множество мономеров использовался один консенсусный мономер, центромера разбивалась на $M-$блоки и дальше блоки кластеризовались([2]). Хотя этот подходит и позволил вывести многие человеческие центромеры в этом подходе центромера не обязательно будет разрешена, особенно в случаях кластеров с большим радиусом. Ниже я описываю простой алгоритм генерации мономеров(MonomerGenrator algorithm), который приближенно решает задачу выделения мономеров.

1.3. Алгоритм генерации мономеров

В дополнение к строке $Centromere$, MonomerGenerator имеет два дополнительных параметра: пороговое значение $maxResolvedDivergence$ и строку $InitialMonomer$(обратите внимание на разницу с параметрами в определение задачи выделения мономеров). Это итеративный алгоритм, который на каждой итерации увеличивает множество мономеров и начинает с множества мономеров, которое состоит из единственной строки $InitialMonomer$. В случае человеческого генома, $InitialMonomr = ConsensusMonomer$, где $ConsensusMonomer$ соответствует консенсусу всех мономеров в человеческом геноме и определен в разделе 1.1.

Для заданной строки $Centromere$ и мономерного множества $Monomers$, $MonomerGenerator$ запускает $StringDecomposer$ для генерации множества блоков $Blocks(Centromere, Monomers)$ и строит граф-блоков, где вершинами графа являются неразрешенные блоки, а ребра соединяют два блока, если расхождение между ними не привышает $\frac{maxResolvedDivergence}{2}$. В связи с тем, что в человеческом геноме порядка 300 000 блоков, построение графа в явном виде стало бы узким местом. Оптимизации для работы с графом описаны в разделе 1.4.

$MonomerGenerator$ выбирает наибольшую связанную компоненту(компоненту с наибольшим количеством вершин) в построенном графе-блоков и вычисляет консенсус мономера $newMonomer$, вычисляя множественное выравнивания всех блоков в компоненте с помощью ClustalOmega([12]). После этого $MonomerGenerator$ расширяет текущее множество мономеров добавляя в него $newMonomer$ и итерируется до тех пор, пока множество мономеров не будет разрешать центромеру $Centromere$. Так же на каждой итерации из множества мономеров удаляются те мономеры, которые не являются наиболее близкими ни к какому из блоков в $Blocks(Centromere, Monomers)$.

Перед тем, как запускать следующую итерацию, MonomerGenerator обновляет последовательности для каждого из мономеров в множестве мономеров. Для обновления мономера ищется консенсус всех блоков

которые разрешают данный мономер. В разрешенной центромере M-консенсус совпадает с соответствующим мономером M в сгенерированном множестве. Не смотря на то, что финальное множество мономеров не является гарантировано $Length - uniform$ для человеческих центромер это не является проблемой, по скольку в человеческом геноме мономеры имеют достаточно консервативную длину равную примерно 171.

1.4. Оптимизация алгоритма

Множество блоков $Blocks(Centromere, Monomers)$ для всего человеческого генома содержит около 300 000 блоков. Если строить граф блоков в явном виде перебором(тогда для каждой пары блоков нужно будет посчитать парное выравнивание), то это будет бутылочным горлышком и по потребляемой памяти и по времени работы. Для ускорения и уменьшения потребляемой памяти я использовала следующий метод.

Для заданной вершину v в графе блоков я считаю расстояние от неё до всех остальных вершин в графе, дальше я использую эту информацию для ускорения поиска компоненты связности в которой находится задана вершина(будем обозначать $component(v)$). Пусть у нас есть список всех вершин в порядке возрастания расстояния до v, тогда мы можем быстро сгенерировать $component(v)$ начиная с вершины v и далее добавляя новые вершины, сканируя этот список. Заметим, что $d(w, u) < \frac{maxResolvedDivergence}{2}$ для любых двух соседних вершин в графе блоков. Более того, если вершина w уже добавлена в $component(v)$, то каждый сосед u в графе блоков удовлетворяет следующему неравенству $d(v, w) + \frac{maxResolvedDivergence}{2} > d(v, u)$. Таким образом нам необходимо только проанализировать вершины, которые находятся на расстояние меньше, чем $d(v, w) + \frac{maxResolvedDivergence}{2}$ в момент, когда мы расширяем $component(v)$ сканируя соседей w. Данное наблюдение дает значительное ускорение при построение графа блоков.

На каждой итерации алгоритма при построение компонент связности блок-графа я случайным образом выбираю вершину v, которая ещё

не принадлежит ни какой компоненте и строю $component(v)$, после этого удаляя данные вершины из дальнейшего рассмотрения.

1.5. Псевдокод и анализ сложности

Сложности времени работы алгоритма складывается из двух основных частей: время работы $StringDecomposer$ и поиск максимальной компоненты в графе блоков.

Время работы StringDecomposer

Время работы $StringDecomposer$ это $O(length(Centromeres) \cdot length(Monomers))$, где $length(String)$ – это суммарная длина всех строк в множестве строк $Strings([11])$. Для человеческого генома это $length(Centromeres) \sim 70Mbp$, $length(Monomers) \sim 400 \cdot 171 = 68,400bp$(на последней стадии алгоритма).

Время построения графа блоков

Поскольку вершины графа – это мономерные блоки, а мономеры имеют достаточно консервативную длину, то сложность построения графа $O(|MBlocks|^2 \cdot |Monomers|^2)$. Для человеческого генома $|MBlocks| \sim 400,000$, $|Monomers| \sim 171bp$.

В худшем случае количество итераций равно $|Monomers| \sim 400$. Сложность наивной реализации $MonomerGenerator$:

$$O(|Monomers| \cdot (length(Centromere) \cdot length(Monomers)) + |MBlock|^2 \cdot |Monomer|$$

Это сложность работы наивной реализации. В разделе 1.4 описаны эвристики, которые позволяют ускорить фактическое время работы алгоритма.

Algorithm 1: MonomerGenerator

Data: Множество строк $Centromere$, строка $InitialMonomer$ и целое число $maxResolvedDivergence$
Result: Множество строк $Monomers$
$Monomers \longleftarrow$ множество строк, состоящие из единственной строки $InitialMonomer$
while $Monomers$ *не разрешает* $Centromere$ **do**
  // Запускаем $StringDecomposer$ что бы получить все блоки в $Centromeres$
  $AllBlocks \longleftarrow Blocks(Centromeres, Monomers)$
  for $M \in Monomers$ **do**
    // сохраняем разрешённые $M-$блоки в множество $MBlocks$
    $MBlocks \longleftarrow$ $ResolvedBlocks(AllBlocks, M, maxResolvedDivergence)$
    if $MBlocks$ *is empty* **then**
      удаляем мономер M из $Monomers$
    else
      // обновляем консенсус у мономера M
      $M \longleftarrow Consensus(MBlocks)$
    end
  end
  // находим неразрешенные блоки
  $UBlocks \longleftarrow$ $UnresolvedBlocks(AllBlocks, Monomers, maxResolvedDivergence)$
  // создаем граф блоков, где каждая вершина соответствует неразрешенному блоку и ребро соединяет похожие блоки
  $BlockGraph \longleftarrow$ граф с множеством вершин из $UBlocks$ и пустым множеством ребер
  for $v, w \in UBlocks$ **do**
    if $EditDistance(v, w) < \frac{maxResolvedDivergence}{2}$ **then**
      добавляем ребро между v и w в $BlockGraph$
    end
  end
  $Component \longleftarrow$ максимальная компонента в $BlockGraph$
  $NewMonomer \longleftarrow$ консенсус всех блоков в $Component$
  добавляем $NewMonomer$ в $MonomerSet$
end
return $MonomerSet$

1.6. Выделение гибридов

Для заданной строки назовём строку состоящую из первых(последних) i нуклеотидов $i-$префиксом(i-суффиксом). Я называю мономер *гибридом* если его можно представить как конкатенацию $i-$префикса мономера X и $j-$суффикса мономера Y. Будем обозначать такой гибрид как $X(i)+Y(j)$, или более кратко $X+Y$, в случаях когда можно опустить индексы i и j. Гибридные мономеры, которые бывают весьма редкие, были обнаружены в нескольких человеческих центромерах([11]). Для каждого нечастого мономера M, $MonomerGenerator$ находит находит наиболее близкого кандидата на гибрида, сгенерированого для каждой пары частых мономеров (X, Y). Я считаю мономер M гибридом, если $div(M, X+Y)$ не превышает $MaxHybridDivergence$(значение по умолчанию 1%)

1.7. Сдвиг множества мономеров

Повторяющийся фрагмент тандемного повтора определен только с точностью до циклического сдвига. Например, $AGGT$, $GGTA$, $GTAG$ и $TAGG$ соответствует 4 разным циклическим сдвигам для тандемного повтора $\cdots AGGTAGGTAGGT\cdots$

Однако, в случае вложенного тандемного повтора, ситуация более сложная. Например, следующий вложенный тандемный повтор $\cdots AGGTAACTTGGTAGGTAACTTGGT\cdots$ состоит из трех "похожих"мономеров AGGT, AACT, TGGT (которые все вместе образуют HOR AGGTAACTTGGT). Сдвиг стартовой позиции у этих мономеров приводит к образованию нового множества мономеров: GTAA, CTTG и GTAG. Заметим, что сдвинутые мономеры не являются циклическим сдвигом оригинальных, а являются гибридами изначальных мономеров. Более того, информации об изначальных мономеров не достаточно, что бы сгенерировать сдвинутое множество мономеров. Более того, в случае центромер с несколькими HOR необходимо знать информацию обо всей центромере, что бы сгенерировать сдвинутое множество мономеров.

К сожалению, в разных работах, где изучались человеческие центромеры использовались разные сдвиги мономеров([4], [7], [6], [16]), из-за этого становится проблематичным сравнивать результаты в разных работах между собой. Эта проблема подчёркивает важность выбора стандартной модели для представления мономеров. Что бы справиться с проблемой сравнения разных множеств мономеров(зачастую с разным начальным сдвигом) у $MonomerGenerator$ есть модуль $MonomerGraph$, который принимает на вход множество мономеров, центромеру и генерирует сдвинутое множество мономеров.

1.8. Мономерные графы

Пусть дана строка $Centromere$ и множество мономеров $Monomers$, $StringDecomposer$ преобразует центромеру в строку $monoCentromere$ над алфавитом из мономеров и символа «?», который обозначает немономерный блок([11]). Мономерный граф – это взвешенный ориентированный граф, вершинами которого являются мономеры. В мономерном графе, две вершины соединяются ребром, если соответствующие им мономеры идут подряд в $monoCentromere$. Вес ребра (M, M') в мономерном графе – это количество раз, когда M' следует за M в $monoCentromere$.

Для заданного мономерного графа, построенного на множестве $Monomers$, $MonomerGraph$ генерирует новое, сдвинутое на i множество мономеров $Monomers(i)$, сдвигая начало каждого из мономеров на i нуклеотидов. Каждое ребро (M, M') в мономерном графе соответствует мономеру $M + M'$ образованному как конкатенация $i-$суффикса мономера M и $j-$префикса мономера M', где $j = |M'| - i$. Однако, поскольку разные ребра могут приводить к одинаковым(или очень похожим) сдвинутым мономерам, я склеиваю два сдвинутых мономеров в один, если расхождение между ними не превышает пороговое значение $\frac{maxResolvedDivergence}{2}$.

$MonomerGraph$ так же строит мономерный граф для сдвинутых мономеров, добавляя ребро между сдвинутыми мономерыми $M + M'$ и

$M'+M''$ для каждого триплетам M, M', M'' в моноцентромере(вес ребра соответствует тому, скольку раз триплет встречается в моноцентромере.)

2. Анализ выделенных мономеров

2.1. Генерация множества мономеров для центромеры X

В таблице 2 находится информация о 23 мономерах выделенных с помощью $MonomerGenerator$ для центромеры X. 12(11) из этих мономеров – частые(не частые), и все кроме двух мономеров – редкие.

Я буду называть *референсными мономерами* мономеры из 6(8, X) центромер, выделенные вручную в работе [11]. В референсных мономерых в 6(8, X) центромере 18(15, 12) мономеров.

Изначальный сдвиг мономеров определяется цикилическим сдвигом $InitialMonomer$, который я взяла из работы [4]. Выравнивание мономеров показало, что референсные мономеры сдвинуты относительно сгенерированных. После сдвига сгенерированных мономеров на 94 я получила 12 частых мономеров и 16 нечастых мономеров(рисунок 3). Частые мономеры соответствуют референсным мономерам, которые составляют канонический DXZ1 HOR в cenX([23]). Нечастые мономеры включают в себя 9 гибридов и 7 вариаций частых мономеров. 11 из 16 мономеров – редкие.

В таблицах 3 и 4 сравниваются мономеры, сгенерированные с помощью $MonomerGenerator$ и референсные мономеры и аннотированы гибриды в центромере X. Нечастые мономеры $W-C$ и $Q-C$ скорее всего являются вариациями референсного мономера B и они отличаются от мономера B крупной делецией в 19 и 33 нуклеотида соответственно.

Частые сгенерированные мономеры очень близки к референсным мономерам. 3 частых мономера полностью совпадают с референсными мономерами, 2 мономера отличаются на единственную вставку, один мономер отличается на единственную замену и 6 мономер отличаются небольшим количеством пропусков либо в начале либо в конце. Несколько замен можно объяснить либо неточностью в референсных мономерах либо полиморфизмом в центромерах внутри популяции. А вставки или удаления на началах и концах объяснимы в небольших

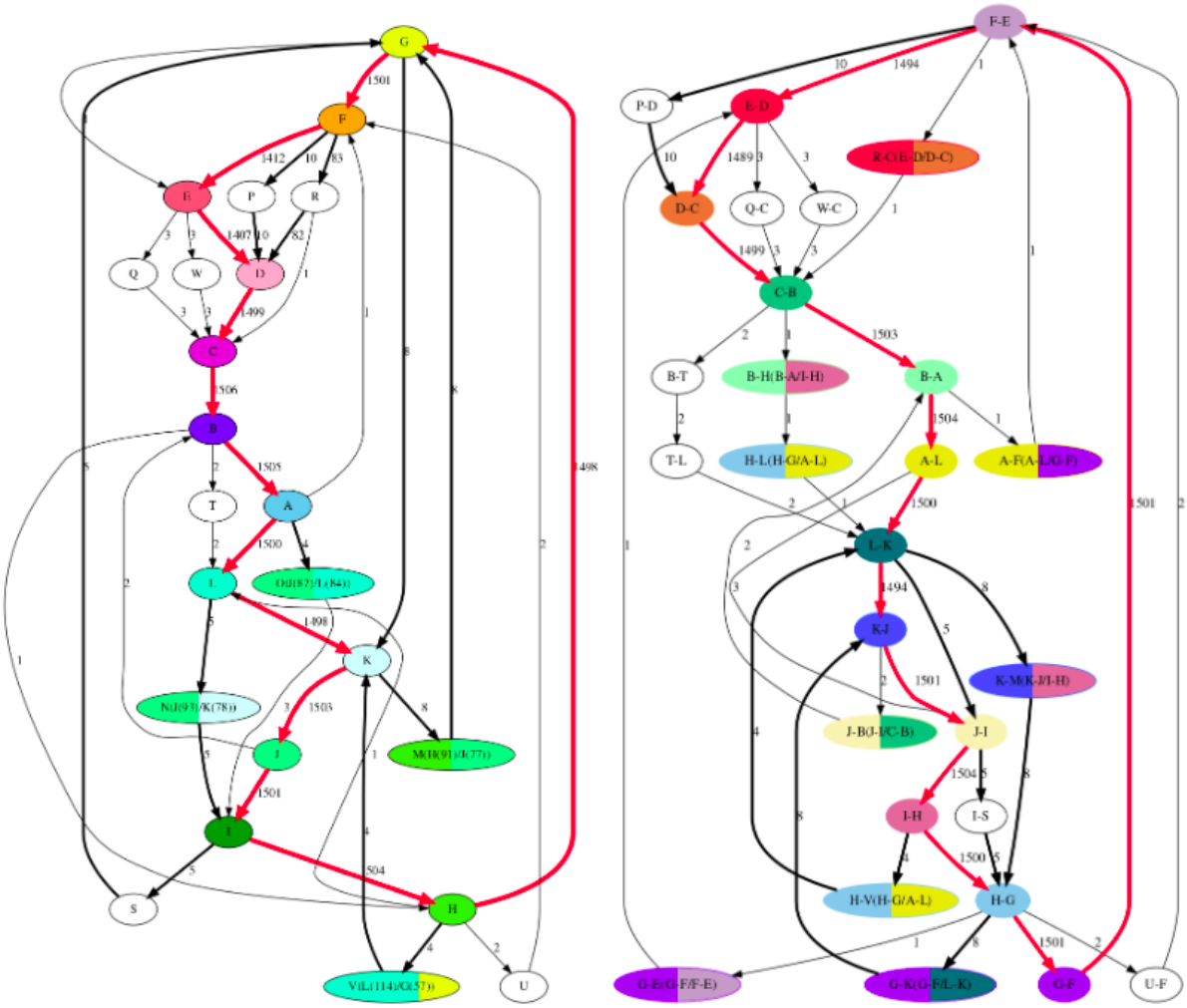


Рис. 3: **Граф мономеров для центромеры X для изначально сгенерированных мономеров(слева) и для сдвинутых мономеров(справа)**. Красный цикл соответствует наиболее частому 12-мономерному HOR в центромере X(DXZ1). Ребра, чей вес превышает 3 показаны жирным. (Слева) мономерный граф состоит из 23 вершин и 41 ребра. (Справа) после сдвига на 94 нуклеотида, мономерный граф состоит из 28 вершин и 44 ребер. Вершины, которые относятся к 12 частым мономерам покрашены в разные цвета. Гибридные мономеры X/Y раскрашены сразу в два цвета, где первый цвет относится к мономеру X, а второй к Y. Обозначение X-Y на графе справа означает, что мономер был получен путем сдвига изначальных мономеров X и Y.

итерация	#разрешённых блоков	#не разрешённых блоков	#не мономерных блоков	размер наибольшей связной компаненты	радиус	сепарация	длина мономера
0	0	18108	37	1507	8	32	171
1	1507	16601	37	1506	8	47	171
2	3015	15093	36	1505	6	35	171
3	4521	13587	37	1501	9	26	171
4	6023	12085	37	1499	7	44	171
5	7525	10583	37	1500	7	30	171
6	9029	9079	37	1499	7	23	171
7	10533	7575	37	1498	9	42	167
8	12036	6072	37	1497	8	30	171
9	13536	4572	37	1496	6	39	186
10	15035	3073	37	1494	8	31	167
11	16535	1573	37	1490	9	25	169
12	18032	76	37	8	1	14	168
13	18040	68	37	5	0	17	171
14	18045	63	38	3	0	22	171
15	18048	60	39	3	1	9	169
16	18052	56	39	3	1	24	163
17	18055	53	39	2	0	9	168
18	18057	51	39	2	0	9	167
19	18059	49	40	2	0	12	167
20	18061	47	40	2	0	9	171
21	18063	45	39	2	3	11	171
22	18065	43	39	2	4	21	171
23	18067	41	38	1	-	-	-

Таблица 2: **Информация о результатах MonomerGenerator на центромере X** Каждая строка соответствует итерации алгоритма. На каждой итерации в множество мономеров добавляется консенсус блоков из максимальной компоненты связности в графе блоков. На нулевой итерации множество мономеров состоит из единственного $ConsensusMonomer$. В первых трёх колонках указано количество разрешенных, не разрешенных и не мономерных блоков после запуска $StringDecomposer$ на множестве мономеров соответствующем текущей итерации. В данной таблице сепарация – это минимальное расстояние от мономера сгенерированного на текущей итерации до всех ранее сгенерированных мономеров(а не всех мономеров).

неточностях в сдвигах в референсных мономерах и в сдвинутых сгенерированных.

Один из мономеров центромеры X выделенных с помощью *MnomerGenerator*(G-K(G-F/L-K)) соответсвует мономеру M, который является K(68) + F(103) гибридом частых мономеров K и F описанных в [6] и [11]. В разложение центромеры M-блок располагается между $J-$блоком слева и $G-$блоком справа. Поскольку канонический HOR в центромере X это следующий 12-мер ABCDE**FG**HI**JK**L, K+F гибрид скорее всего образован следующей делецией ABCDE**FG**HI**JK**LABCDE**F**G**H**I**JK**L в которой удалены суффикс $F-$блока и преффикс $K-$блока. Аналогично мономер $K - M(K - J/I - H)(H - V(H - G/A - L))$ это гибрид G(129)+I(42)(J(112)+E(55)), который вероятно образован делецией в ABCDEFGHIJKL (ABCDEFGHIJKLABCDEFGHIJKL). Также, *MonomerGenerator* вывел 5 редких гибридных мономеров.

сгенерированные мономеры	референсные мономеры	расхождение с референсными мономерами	количество вхождений в cenX
Частые мономеры			
F-E	L	1.16	1412
E-D	A	1.76	1407
D-C	B	0	1499
C-B	C	0	1506
B-A	D	0.58	1505
A-L	E	0.59	1500
L-K	F	0.58	1498
K-J	G	0.58	1503
J-I	H	0	1501
I-H	I	0.58	1504
H-G	J	0.6	1498
G-F	K	0.58	1503

Таблица 3: **Сравнение (сдвинутых) частых мономеров сгенерированных *MonomerGenerator* и референсных мономеров для центромеры X** В первой колонке указан индетефикатор сгенерированных мономеров после сдвига. Например, мономер, который обозначен $X - Y$ обозначает мономер который получен с помощью сдвига мономера X и Y в изначальном сгенерированном множестве. Во второй колонке указан ближайший мономер из множества референсных мономеров. Третья колонка – расхождение между сгенерированным мономером и ближайшим из референсных. Четвертая колонка – количество M-блоков в центромере X, соответствующих сгенерированному мономеру M. Для каждого сгенерированного мономера в таблице показано глобальное выравнивание с ближайшим референсным мономером(штраф за мисматчи, вставки и удаления – 1, награда за совпадение – 1). Первая(вторая) линия в выравнивание соответствует референсному(сгенерированному) мономеру.

сгенерированные мономеры	референсные мономеры	расхождение с референсными мономерами	количество вхождений в cenX
Не частые мономеры			
G-K(G-F/L-K)	K+F	7.6+6.43	8
P-D	A	4.7	10
K-M(K-J/I-H)	G+I	4.09+19.88	8
I-S	I	4.09	5
H-V(H-G/A-L)	J+E	3.57+10.12	4

Таблица 4: **Сравнение (сдвинутых) не частых мономеров сгенерированных** *MonomerGenerator* **и референсных мономеров для центромеры X** В первой колонке указан индетефикатор сгенерированных мономеров после сдвига. Например, мономер, который обозначен $X - Y$ обозначает мономер который получен с помощью сдвига мономера X и Y в изначальном сгенерированном множестве. Во второй колонке указан ближайший мономер из множества референсных мономеров. Если во второй колонке написано два мономера разделенных ”+ значит сгенерированный мономер является гибридом этих двух. Например, мономер ”G-K”гибрид мономеров K и F([11]). Третья колонка – расхождение между сгенерированным мономером и ближайшим из референсных(или до двух референсных в случае гибрида). Четвертая колонка – количество M-блоков в центромере X, соответствующих сгенерированному мономеру M. Для каждого сгенерированного мономера в таблице показано глобальное выравнивание с ближайшим референсным мономером(штраф за мисматчи, вставки и удаления – 1, награда за совпадение – 1). Первая(вторая) линия в выравнивание соответствует референсному(сгенерированному) мономеру.В случае гибрида участок который относится к каждому из референсных мономеров подсвечен.

2.2. Генерация мономеров для центромер 6 и 8

MonomerGenerator сгенерировал множество мономеров размера 23 и 14 для центромер 6 и 8 соответственно(таблица 5, 6). Для того что бы сравнить сгенерированные мономеры с референсными я сдвинула сгенерированные мономеры на 77 нуклеотидов. В этом разделе я привожу сравнение только частых мономеров с референсными множеством мономеров. В таблице 7(8) приведено сравнение нуклеотидных последовательностей мономеров полученных с помощью *MonomerGenerator* с референсными мономерами для центромер 6(8).

18(5) мономеров в центромере 6 частые(нечастые). После сдвига и склеивания похожих мономеров было получено множество мономеров из 18 частых и 5 нечастых мономеров. Частые мономеры соответствуют референсным мономерам, которые образуют $D6Z1HOR$([11]) и широко представлены в 6 центромере.

11(3) мономера в центромере 8 частые(редкие). После сдвига и склеивания похожих мономеров было получено множество мономеров для 8 центромеры состоящие из 12 частых мономеров и 6 нечастых. Частые мономеры соответствуют референсным мономерам, которые образную D8Z3 HOR([11]) и широко представлены в центромере 8.

В референсном множестве мономеров для 8 центромеры 15 мономеров, но количество частых мономеров в сдвинутом множемтве мономеров сгенерированных *MonomerGenerator* только 12. Некоторые референсные мономеры очень похожи друг на друга, например мономеры D и L отличаются в всего одном нуклеотиде, а мономеры E и M(так же как F и N) в трёх. *MonomerGenerator* сгенерировал один мономер для каждой из этих пар.

Для 6 и 8 центромер частые мономеры по большей части совпадают с референсными мономерами. Некоторые мономеры имеют небольшие пропуски в начале или в конце, что можно объяснить небольшим несоответсвием сдвигов. Некоторые мономеры имеют небольшое количество(в самом плохом случае 6) несовпадений(замен, вставок, удалений) с референсными мономерами, что можно объяснить неточностями

итерация	# разрешённых блоков	# не разрешенных блоков	# не мономерных блоков	размер максимальной компоненты	# удалённых мономеров	радиус	минимальное расстояние до ранее сгенерированных мономеров	длина нового мономера
0	0	16315	0	956	1	5	100	169
1	0	16315	0	956	1	7	42	172
2	0	16315	0	955	1	6	41	171
3	0	16315	0	954	1	5	39	171
4	0	16315	0	954	1	5	29	169
5	0	16315	0	954	1	4	25	171
6	0	16315	0	954	1	6	29	170
7	6687	9628	0	954	0	6	31	170
8	6687	9628	0	953	0	5	16	169
9	6687	9628	0	952	0	4	27	171
10	6687	9628	0	952	0	6	27	169
11	10505	5810	0	953	0	5	16	171
12	10505	5810	0	951	0	7	23	168
13	10505	5810	0	940	0	5	21	171
14	10505	5810	0	669	0	10	18	169
15	10505	5810	0	669	0	5	28	169
16	10505	5810	0	656	0	5	22	167
17	15584	731	0	654	0	4	9	170
18	15584	731	0	45	0	3	8	169
19	16294	21	0	3	0	2	10	171
20	16294	21	0	2	0	0	12	171
21	16294	21	0	2	0	0	31	138
22	16301	14	0	2	0	2	10	163
23	16299	12	0	1	0	-	-	-

Таблица 5: **Информация о мономерах, сгенерированных с помощью** *MonomerGenerator* **для центромеры 6**

в референсных мономерах и полиморфизмом в центромерах внутри популяции.

итерация	**# разрешённых блоков**	**# не разрешенных блоков**	**# не мономерных блоков**	**размер максимальной компоненты**	**# удалённых мономеров**	**радиус**	**минимальное расстояние до ранее сгенерированных мономеров**	**длина нового мономера**
0	0	12239	4	1514	1	7	34	171
1	1516	10709	18	1514	0	7	39	171
2	3030	9204	9	1510	0	8	52	167
3	4544	7695	4	1513	0	8	27	171
4	6044	6195	4	923	0	7	31	171
5	6969	5270	4	916	0	8	23	171
6	7895	4344	4	913	0	7	30	171
7	8808	3431	4	852	0	8	27	167
8	9661	2578	4	851	0	7	25	167
9	10513	1726	4	850	0	8	34	171
10	11366	873	4	844	0	7	27	170
11	12217	22	4	7	0	7	9	171
12	12230	9	4	2	0	4	62	113
13	12233	6	4	2	0	0	11	167
14	12235	4	4	1	0	-	-	-
15	10505	5810	0	669	0	5	28	169
16	10505	5810	0	656	0	5	22	167
17	15584	731	0	654	0	4	9	170
18	15584	731	0	45	0	3	8	169
19	16294	21	0	3	0	2	10	171
20	16294	21	0	2	0	0	12	171
21	16294	21	0	2	0	0	31	138
22	16301	14	0	2	0	2	10	163
23	16299	12	0	1	0	-	-	-

Таблица 6: **Информация о мономерах, сгенерированных с помощью** *MonomerGenerator* **для центромеры 8**

сгенерированные мономеры	референсные мономеры	расхождение с референсными мономерами	количество вхождений в cenX
	Частые мономеры		
B-F	J_9_split_1000964_1004021_doubled/1474_1646/F	4.62	955
F-D	K_10_split_1000964_1004021_doubled/1647_1818/F	2.9	955
D-E	L_11_split_1000964_1004021_doubled/1819_1988/F	2.33	957
E-G	M_12_split_1000964_1004021_doubled/1989_2159/F	1.17	957
G-H	N_13_split_1000964_1004021_doubled/2160_2330/F	1.75	957
H-A	O_14_split_1000964_1004021_doubled/2331_2501/F	1.74	952
A-I	P_15_split_1000964_1004021_doubled/2502_2670/F	1.18	952
I-J	Q_16_split_1000964_1004021_doubled/2671_2842/F	1.16	952
J-K	R_17_split_1000964_1004021_doubled/2843_3011/F	3.53	952
K-L	A_0_split_1000964_1004021_doubled/3012_3183/F	2.33	952
L-M	B_1_split_1000964_1004021_doubled/119_286/F	2.35	937
M-N	C_2_split_1000964_1004021_doubled/287_454/F	1.79	669
N-P	D_3_split_1000964_1004021_doubled/455_623/F	1.77	669
P-O	E_4_split_1000964_1004021_doubled/624_792/F	2.37	669
O-R	F_5_split_1000964_1004021_doubled/793_964/F	1.16	954
R-S	G_6_split_1000964_1004021_doubled/965_1133/F	0.59	954
S-C	H_7_split_1000964_1004021_doubled/1134_1304/F	1.75	954
C-B	I_8_split_1000964_1004021_doubled/1305_1473/F	2.35	954

Таблица 7: **Сравнение сдинутых сгенернированных мономеров с помощью** *MonomerGenerator* **с референсными мономерами в центромере 6.**

сгенерированные мономеры	референсные мономеры	расхождение с референсными мономерами	количество вхождений в cenX
Частые мономеры			
E-F	A	0	925
F-G	B	0	925
G-A	C	1.17	660
A-B	D|L	2.34; 2.92	1513
B-C	E|M	3.5; 2.92	1511
C-D	F|N	1.75; 0	1500
L(D+G)-H	G	0	851
H-I	H	3.52	850
I-J	I	4.68	852
J-K	J	2.34	853
K-A	K	2.34	853
D-E	O	5.08	911

Таблица 8: **Сравнение сдинутых сгенернированных мономеров с помощью** *MonomerGenerator* **с референсными мономерами в центромере 8.**

2.3. Генерация мономеров для всех живых человеческих центромер

В таблице 9 представлены результаты запуска $MonomerGenerator$ на всех человеческих центромерах(смотри раздел 1.1). В итоге было выделено 220 частых мономеров, 33 гибридных и 155 не частых. При запуске $MonomerGenerator$ на всём геноме редкие мономеры не выделялись.

Рисунок 4 показывает распределение радиусов и сепараций для всех частых человеческих мономеров. Я использовала коэффициент сепарации для оценки качества сгенерированных мономеров(для мономеров с высоким коэффициентом сепарации однозначно выделяются из $M-$блоки). Я проанализировала $separationRatio(Centromere)$ и выделила минимальный коэффициент сепарации из всех мономеров для данной центромеры. Например, у центомеры 8 наибольший $separationRatio(cen8) = 1.9$, а для центромеры 1 минимальный $separationRatio(cen1) = 0.1$.

У 12 человеческих центромер коэффицент сепарации не меньше 1. Большинство центромер с коэффицентом сепарации меньше 1 содержат мономеры с нетипично большим радиусом, что может говорить о наличии "старых"мономеров, которые значительно расходятся с консенсусом. Более того, практически все центромеры с коэффициентом сепарации меньше 1(за исключением центромеры 7 и 13) содержат мономеры которые встречаются сразу в нескольких центромерах. Радиус у таких мономеров из нескольких центромер может быть больше радиуса других мономеров, потому что они образованы "субмономерами"из различных центромер(с немного разным консенсусом), которые были сгруппированы вместе при помощи $MonomerGenerator$.

chr	# частых мономеров	# гибридных мономеров	общее # мономеров	макс. радиус	мин. коэффицент сепарации
1	10	7	22	20	0.1
2	4	0	9	13	0.615
3	17	0	23	9	1.5
4	17	3	22	13	0.833
5	8	5	14	20	0.1
6	18	1	19	10	1.286
7	6	1	14	17	0.765
8	11	1	12	10	1.857
9	9	5	23	13	0.615
10	8	6	36	12	1.11
11	5	1	13	13	1
12	8	5	20	16	0.562
13	10	1	14	10	1.375
14	8	1	12	18	1
15	12	0	16	10	1.125
16	10	0	16	20	0.1
17	16	2	43	14	1.5
18	11	2	20	12	0.89
19	6	2	26	20	0.1
20	15	0	17	13	0.615
21	10	1	14	10	1.375
22	8	1	13	18	1
X	12	2	14	9	1.625
Total	220	33	375	20	0.1

Таблица 9: **Информация о мономерах выделенных с помощью** *MonomerGenerator* **на всём человеческом геноме** Каждая строка соответствует информации о выделенных мономерах для конкретной хромосомы. Вторая(третья) колонка показывает информацию о количестве частых(гибридных) мономеров сгенерированных с помощью *MonomerGenerator* на каждой из хромосом. Четвертая колонка показывает итоговое количество мономеров для данной центромеры включая в себя частые, не частые и гибридные мономеры. Пятая(шестая) колонка показывают максимальный радиус(минимальный коэффецент сепарации) для частых мономеров в данной хромосоме. Строки, с коэффицентом сепарации меньше(не меньше) 1 выделены красным(зелёным).

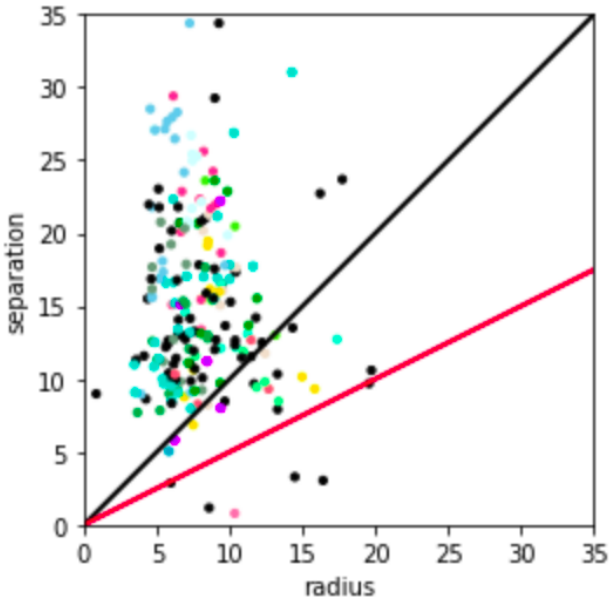


Рис. 4: **Информация о радиусе(ось x) и о сепарации(ось y) для всех 220 частых мономеров.** Только 21 из 220 частых мономеров находиться под чёрной линией $separation = radius$, в этом случае коэффициент сепарации меньше 1(и только 4 из них располагаются под красной линией $separation = 0.5 \cdot radius$, у них коэффициент сепарации меньше 0.5). Каждый круг соответствует частому мономеру. Для более понятной картины каждый круг немного смещён случайным образом. Цвет точки соответствует хромосоме, кроме черных точек, которые соответствуют мономерам, которые находятся сразу на нескольких центромерах. Светло розовый круг с радиусом 10 и сепарацией 1 соответствует центромере 1.

2.4. Сравнение мономеров полученных с помощью $MonomerGenerator$ и T2T мономеров

T2T мономеры

Для изучения центромер в $T2T$ консорциуме используются мономеры полученные в ручную в рамках работ [4] и [7]. Будем обозначать это множество мономеров как $MonomersT2T$

Выделение только частых мономеров

$MonomerGenerator$ выделяет в том числе и нечастые мономеры, таким образом множество мономеров может быть сильно больше, чем количество мономеров в $MonomersT2T$, это затрудняет сравнение двух множеств мономеров.

Количество вхождений мономера M в центромеру(обозначим как $count(M) = count(M, Centromere^*)$) определим как количество вхождений M в моноцентромеру $Centromere*$. Отсортируем все мономеры по убыванию количества и обозначим $i-$ый по частоте мономер как M_i. Для заданного множества $Monomers_i = \{M_1 \cdots M_i\}$ i наиболее частых мономеров определим $count(Monomers_i)$ как суммарное количество всех вхождений мономеров из множества. Определим $imin$ как минимальное значение i, такое что $count(Monomers_i)$ превышает пороговое значение $MinFraction \cdot |Centromere^*|$, где $|Centroemre^*|$ – это длина моноцентромеры(значение по умолчанию $MinFraction = 0.9$). Построим множество частых мономеров, как $Monomers_{imin}$ дополненное мономерами $M_{imin+1}, M_{imin+2}, \cdots$ пока количество новодобавляемого мономера не меньше, чем $MinExtension \cdot count(M_{imin})$(значание по умолчанию=0.7). Получившиеся итоговое множество мономеров будем обозначать, как $MonomerNew$.

Метод сравнения двух мономерных множеств

Для заданного множества мономеров $Monomers$ я разделяю центромеры $Centromere$ на мономерные блоки $Blocks$. Я определяю *средний*

радиус мономера M(обозначим $r(M)$) как среднее расстояние между мономером M и всеми $M-$блоками в $Blocks$. Для двух строк S' и S'' обозначим редакционное расстояние между ними как $distance(S', S'')$.

С точки зрения кластеризации множество мономеров – это центры кластеров множества мономерных блоков. Я использовала среднее квадратичное отклонение и $Davies - Bouldin$ индекс([10]) для оценки качества кластеризации.

Среднее квадратичное отклонение определяется следующим образом:

$$distortion(Monomers, Blocks) =$$

$$\frac{1}{|Blocks|} \cdot \sum_{M \in Monomers} \sum_{M-blocks Block \in Blocks} distance(M, Block)^2$$

$Davies - Bouldin$ индекс определяется как

$$DBI(Monomers, Blocks)$$

$$\frac{1}{|Monomers|} \cdot \sum_{M \in Monomers} \max_{M' \neq M} \frac{(r(M) + r(M'))}{distance(M, M')}$$

Множество мономеров $MonomersNew$ как правило содержит больше мономеров, чем множество $MonomersT2T$ для всех центромер, кроме 3, 4, 10, 13, 17, 18, 20 и 21. Поскольку для сравнения двух вариантов кластеризации для одного множества точек, необходимо, что бы количество кластеров было одинаковым, я выбираю подмножество $MonomersNew$, такой, что бы размер подмножества был такой же как размер $MonomersT2T$. Для каждого мономера из $MonomersT2T$ мы выбираем ближайший мономер из $MonomersNew$ и строим множество $MonomersNew^*$ такого же размера, как и множество $MonomersT2T$ (в этом случае $MonomersT2T^* = MonomersT2T$). Аналогично, если наоборот множество $MonomersT2T$ содержит больше мономеров, чем $MonomersNew$, то для каждого мономера из $MonomersNew$ мы находим ближайший мономер в $MonomersT2T$ и строим множество мономеров $MonomersT2T^*$ такого же размера как $MonomerNew$(в этом случае $MonomersNew^* = MonomersNew$).

Сравнение $MonomersNew$ **и** $MonomersT2T$

В таблице 10 сравниваются множества мономеров $MonomersT2T^*$ и $MonomersNew^*$. Для того, что бы убедиться что сравнение является адекватным(то есть сравнивается кластеризация для одинакового множества точек), я определяю множество мономерных блоков $SharedBlocks$, которое является общим для обоих множеств мономеров. Для каждого мономерного блока B'(который был получен в результате декомпозиции центомеры на мономеры из $MonomersNew^*$) и для каждого пересекающегося мономерного блока B''(который был получен в результате декомпозиции центромеры на мономеры из $MonomersT2T^*$), новые блок B получен как пересечение B' и B''. Я добавляю B в множество $SharedBlocks$ если его длина превышает пороговое значение $MinSharedLength$(значение по умолчанию $MinSharedLength = 150$). Поскольку блок B из $SharedBlocks$ обычно немного короче, чем блоки B' или B'', я изменила определение расстояния между B и мономером M на минимальное расстояние между B и подстрокой $M-$консенсуса, где $M-$консенсус это консенсус всех множественных выравниваний $M-$блоков.

Таблица 10 показывает, что множество $MonomersNew^*$ является более хорошей кластеризацией мономерных блоков, чем $MonomersT2T^*$(если смотреть на среднее квадратичное отклонение). У $MonomersT2T^*$ лучше среднее квадратичное отклонение для центромер 15 и 21.

cen	# мономеров в Monomers-T2T*/ Monomers-New*	# блоков в Monomers-T2T* / Monomers-New* / SharedBlocks	среднее квадратичное отклонение Monomers-T2T* / Monomers-New*	Davies-Bouldin индекс для MonomersT2T* / MonomersNew*	# мономеров из Monomer-Generator	# мономеров в Monomers-T2T	# мономеров в Monomers-New
1	6	26504/26504/26486	3.69/**3.27**	**2.92**/3.35	24	6	11
2	4	13744/13744/13744	2.59/**2.29**	1.99/**1.96**	9	4	4
3	17	8502/8502/8502	1.92/**1.66**	1.94/**1.82**	23	23	17
4	17	21715/21715/21711	1.92/**1.52**	**2.17**/2.23	23	19	17
5	6	14893/14893/14893	4.29/**3.63**	**1.75**/2.05	15	6	8
6	18	16315/16315/16313	1.21/**0.87**	1.53/**1.47**	19	18	18
7	6	19375/19375/19369	1.39/**1.30**	4.09/**4.02**	14	6	6
8	11	12246/12246/12242	1.53/**1.30**	0.99/**0.98**	12	11	12
9	7	15456/15456/15456	3.87/**3.83**	**2.29**/2.32	23	7	9
10	10	13243/13243/13243	3.02/**2.71**	3.40/**2.35**	36	24	10
11	5	19872/19872/19872	2.24/**1.79**	1.06/**1.01**	13	5	5
12	8	15204/15204/15204	1.99/**1.86**	2.82/**2.70**	20	8	8
13	10	11539/11539/11537	1.42/**1.06**	1.74/**1.73**	14	20	10
14	8	15349/15349/15349	**1.49**/**1.49**	**2.55**/2.57	12	8	8
15	11	5961/5961/5961	**1.80**/1.94	**2.30**/2.35	16	11	12
16	10	11643/11643/11641	1.77/**1.42**	3.71/**3.25**	19	10	10
17	31	24345/24345/24342	1.86/**1.49**	**2.59**/2.61	43	44	31
18	10	29285/29285/29285	2.30/**1.83**	2.64/**2.54**	20	12	10
19	6	23220/23220/23219	3.37/**2.85**	**3.68**/10.26	25	6	6
20	15	12793/12793/12785	1.62/**1.52**	**2.14**/2.29	18	16	15
21	10	2021/2021/2021	**1.76**/2.03	**2.17**/2.22	14	19	10
22	8	17146/17146/17146	1.62/**1.57**	2.85/**2.76**	13	8	8
X	12	18108/18108/18108	1.56/**1.44**	1.71/**1.70**	14	12	12

Таблица 10: **Сравнение множества мономеров выделенных с помощью** $MonomerGenerator$ **и ранее выделенных мономеров T2T консорциума** Номер центромеры(первая колонка), количество мономеров в множестве $MonomersT2T^*$ и $MonomersNew^*$(вторая колонка), количество блоков в $MonomersT2T^*/Monomers^*/SharedBlocks$(третья колонка), среднее квадратичное отклонение для $MonomersT2T^*$ и $MonomersNew^*$(четвертая колонка), Davies-Bouldin индекс для множества мономеров $MonomersT2T^*$ и $MonomersNew^*$(пятая колонка), количество мономеров полученных с помощью $MonomerGenerator$(шестая колонка), количество мономеров в множестве $MonomersT2T$(седьмая колонка) и количество мономеров которые соответствуют частым мономерам в выделенном множестве(восьмая колонка)

3. Выделение канонического HOR

3.1. Ошибочно склеенные и расклеенные мономеры

Какое именно множество мономеров с точки зрения биологии является оптимальным остается неясным. Не смотря на это, если рассматривать мономеры с точки зрения их расположения в моноцентромере(например, пары(тройки) подряд идущих мономеров), то можно заметить, что какие-то из мономеров были ошибочко склеены(разделены). И если рассматривать ещё и информацию о позициях, то можно получить более адекватное множество в соответствие с CE постулатом. Будем называть два мономера *похожими*, если процент идентичности меду ними не превосходит $minPI$(значение по умолчанию 94%).

Рассмотрим два мономера M' и M'' которые соответствуют двум наиболее *похожим* мономерам в множестве мономеров $Monomers$ и если бы параметры для выделения мономеров были бы немного другие, то эти бы два мономера были бы склеены вместе. Поскольку не ясно каким образом стоит подбирать параметры для кластеризации, так же не ясно, что является более разумным объединить эти два кластера в один или наоборот. Однако, если мономеры M' и M'' всё время окружены одинаковыми мономерами X и Y в моноцентромере(то есть часто встречаются триплеты $XM'Y$ и $XM''Y$), то скорее всего эти два мономера были ошибочно расклеены и их следует объединить вместе в один мономер M, который будет определен как консенсус всех M'-блоков и M''-блоков. Такая склейка хорошо соотносится с CE постулатом в рамках которого ни один не гибридный мономер не может встречаться в HOR более одного раза. Поэтому, если мономеры M' и M'' не склеены, нельзя представить HOR в виде, когда он проходит через X и Y ровно один раз, а это необходимо в рамках CE постулата.

С другой стороны, если обычно мономер M встречается в контексте X' и Y'(то есть в моноцентромере есть триплет $X'MY'$) и в контексте X'' и Y'', то это противоречит CE постулату. Скорее всего в этом случае

два разных мономера ошибочно слеены в один и их нужно разделить на два мономера M' и M'', таких что, у одного контекст будет X' и Y', а у второго X'' и Y''. Мономер $M'(M'')$ можно определить, как консенсусы всех M'-блоков(M''-блоков) в триплетах $X'M'Y'(X''M''Y'')$.

При этом такое преобразование не обязательно будет приводить к более оптимальной кластеризации. Множество мономеров полученное после операций слития и расклейки множетсва мономеров $MonomerNew$ будем обозначать как $MonomerNew^{+}$.

3.2. Мономеры, похожие по позициям

Для заданного мономера M из множества мономеров $Monomers$ для заданной моноцентромеры, я нахожу все тройки мономеров XMY, которые идут подряд в заданной моноцентромере. Далее я строю матрицу триплетов $Triples_M$ размера $|Monomers| \times |Monomers|$, где $Triples_M(X, Y)$ – это количество раз, сколько триплетXMY встречается в моноцентромере. Далее я строю нормализованную матрицу триплетов $NormalizedTriples_M(X, Y)$, которая получена из метрицы $Triples_M$ на такую константу, что бы сумма всех элементов в матрице была равна 1.

Для двух матриц A и B размера $N \times M$, я определяю *похожесть* как скалярное произведение векторов размера $N \cdot M$ соответствующих матрицам:

$$sim(A, B) = \sum_{i,j} A(i, j) \cdot B(i, j)$$

Для двух мономеров M и M' я определяю *похожесть по позициям* $PosSim(M, M')$ как

$$similarity(NormalizedTriples_M, NormalizedTriples_{M'})$$

Будем называть два мономера *похожими*, если процент идентичности меду ними не превосходит $minPI$(значение по умолчанию 94%). Будем говорить, что два похожих мономера ещё и *похожи по позици-*

ям, если их *похожесть по позициям* превышает пороговое значение $minPosSim$(значение по умолчанию 0.4)

3.3. Склейка и разделение мономеров

Поскольку два мономера похожих по позициям скорее всего были ошибочно расклеены, $HORmon$ проверяет есть ли в множестве $Monomers$ пары мономеров похожих по позициям. Если такие пары есть, то алгоритм итеративно находит пары мономеров наиболее схожим по позициям(похожим по позициям, у которых похожесть по позиции наибольшая), склеивает их в новый мономер, добавляет в множество мономеров новый мономер и запускает $StringDecomposer$ с новым(меньшим) множеством мономеров, и итерация продолжаются, пока не закончатся все мономеры близкие по позициям. Это процедура гарантировано сходится, поскольку с каждой итерацией множество мономеров уменьшается, а количество мономеров конечно. Аналогично тому как $HORmon$ строит матрицы для триплетов XMY,$HORmon$ так же строит матрицы для триплетов XYM и MXY и склеивает мономеры на основе этих матриц аналогичным образом.

Для того, что бы расклеить мономер M, $HORmon$ анализирует все частые триплеты XMY в моноцентромере. Для заданного мономера M, я обозначаю максимальный элемент в матрице $NormalizedTriples_M(X, Y)$ как M-лидер. Я классифицирую пары (X, Y) и (X', Y') из матрицы $NormalizedTriples_M(X, Y)$ как $M-$сравнимые если

$$\frac{NormalizedTriples_M(X', Y')}{NormalizedTriples_M(X, Y)}$$

превосходит пороговое значение для расклейки $splitValue$(значение по умолчанию $\frac{1}{8}$). HORmon использует singke linkage кластеризацию для поиска всех пар мономеров (X, Y), которые $M-$сравнимы с $M-$лидером и обозначает их как $M-$пары-кандидаты.

Две мономерные пары (X, Y) и (X', Y') называются независимыми, если все четыре мономеры X, Y, X', Y' разные. Назовём моно-

мера M у которого есть $M-$пары-кандидаты *разделяемым* если все $M-$пары-кандидаты попарно независимы и *неразделяемым* в ином случае. Пусть у нас есть *разделяемый* мономер M, $HORmon$ рассматривает все $M-$пары-кандидаты $(X_1, Y_1), \cdots, (X_t, Y_t)$ и разделяет мономер M на t мономеров $M_1, \cdots, M_t$, где M_i – это консенсус всех $M-$блоков, которые входят в состав триплета $X_i M Y_i$ внутри моноцентромеры.

Algorithm 2: SplitAndMerge. Псевдокод модуля $HORmon$ для склейки и расклейки мономеров

```
Data: Centromere, Monomers, minPI, minPosSim, splitValue
Result: Множество мономеров Monomers после склейки и
        расклейки
while существуют похожие по позициям мономеры из
 Monomers(относительно minPI и minPosSim) do
    находим самые близкие по позициям мономеры M' и M'' из
     Monomers
    выводим новый мономер M как консенсус всех M'−блоков и
     M''−блоков в Centromere
    удаляем мономеры M' и M'' из Monomers
    добавляем M в Monomers
end
for разделяемый M ∈ Monomers(относительно splitValue) do
    for M-пары-кандидата (X, Y) do
        находим новый мономер M' как консенсус всех
         M−блоков в контексте XMY
        добавляем M' в Monomers
    end
    удаляем M из Monomers
end
return Monomers
```

3.4. И ещё раз про мономерные графы

Понятие мономерного графа определено в 1.8. Сейчас я определю это понятие ещё раз, на этот раз с использованием моноцентромеры. Пусть нам дана моноцентромера $Centromere^*$. Мономерный граф – это взвешенный ориентированный граф, вершинами которого являются мономеры и две вершины, которые соответствуют мономерам M и M' соеди-

нены ребром, если последовательность MM' встречаются в моноцентромере $Centromere^*$. Вес ребра – это количество раз, которое подстрока MM' входит в моноцентромеру. Заметим, что мономерный граф – это де Брюин граф для k=2 $DB(Centromere^*, 2)$([9]). На рисунке 5(сверху) показан мономерный граф для центромеры X для T2T генома, в котором есть 12 ребер с высоким весом, которые образуют канонический HOR и в добавок на рисунке показано два редких гибридных мономера(весом 5 и 8) и 15 ребер маленького веса.

На рисунке 5(снизу) показан мономерный граф для центромеры X для HPRC генома, в котором точно так же жирный цикл составляет канонический HOR в центромере X. По сравнению с центромерой X в T2T геноме, центромера X в $HPRC$ геноме содержит всего один из двух гибридов, что может свидетельствовать, что гибридные мономеры появились относительно недавно. Так же в графе присутствуют только 5 из 15 ребер с маленьким весом по сравнению с T2T геномом, это иллюстрируют вариативность человеческих центромер внутри популяции.

Рисунок 5 может создать ложное впечатление, что если просто удалить ребра с маленьким весом в мономерном графе, то тогда граф будет представлять из себя просто цикл состоящий из тяжелых ребер и этот цикл и будет соответствовать каноническому HOR. После процедур склейки и расклейки произведенных над множеством мономеров полученных по результатам $MonomerGenerator$ это действительно так для центромер 3, 11, 14, 16, 17, 19, 20, 21, 22 и X. Однако у других центромер более сложная эволюционная архитектура, она проанализирована ниже в следующих разделах.

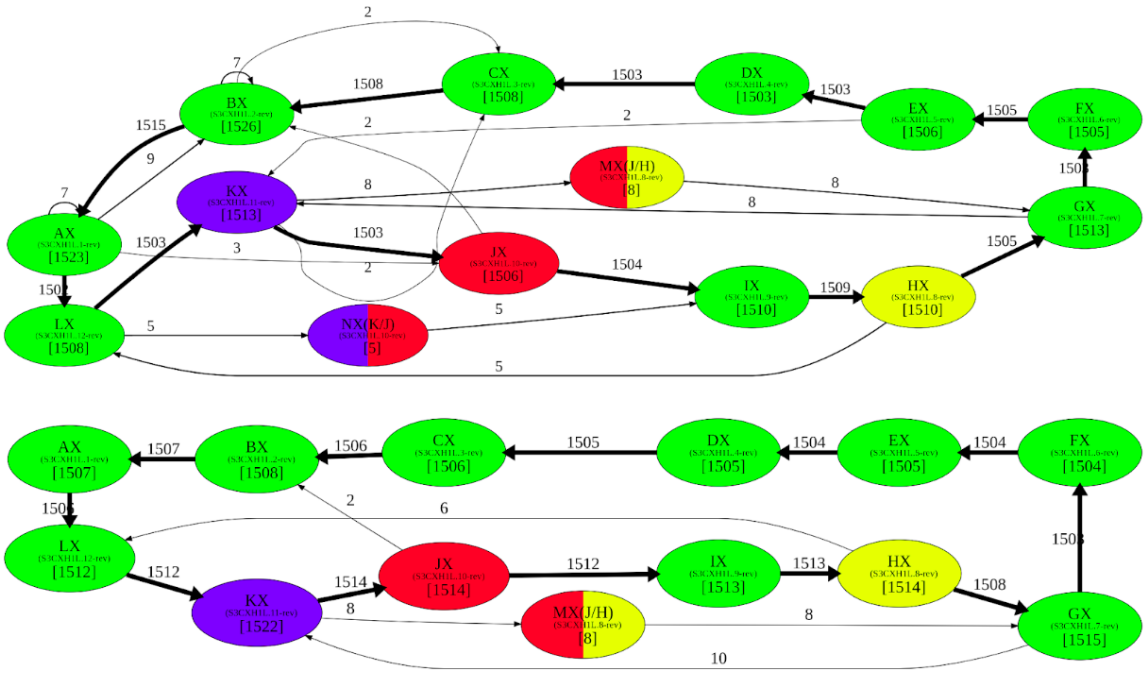


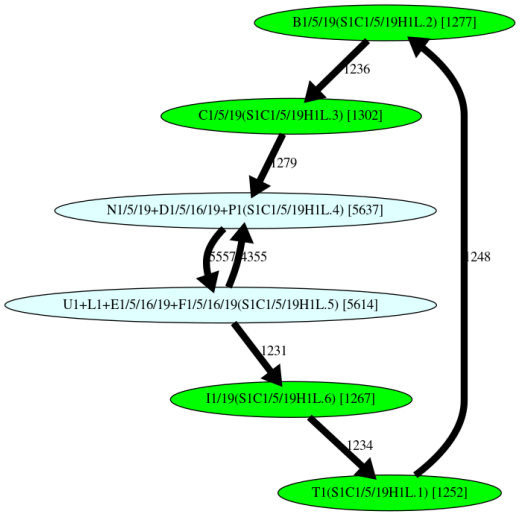
Рис. 5: **Граф мономеров для центромеры X для T2T(сверху) и HPRC(снизу) геномов**. Мономерные графы для центромеры X были построены для множества мономеров состоящих из двух редких гибридов(обозначенных как M и N) и 12 частых канонических мономеров(обозначенных как A, B, $\cdots$, K и L), которые составляют канонический DXZ1 HOR в центромере X(S3CXH1.L [7]). Мелкий шрифт соответствует классическим именам мономеров в соответствие с [7]. Гибридные мономеры были описаны в [11]. Гибридные мономера образованные из мономеров X и Y соответствуют двухцветным вершинам(два цвета соответствуют мономерам X и Y) и в имени записано (X/Y). На графе показаны только ребра веса не меньше 1(ребра с весом хотя бы 100 выделены жирным). Цикл образованный жирными ребрами(с весом хотя бы 1500) состоит из 12 наиболее частых мономеров и образуют канонический HOR

3.5. Мономерные графы для человеческих центромер

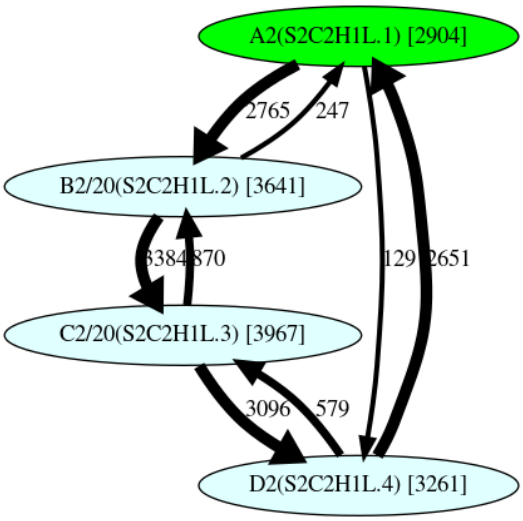
Для заданного множества мономеров $Monomers$ и моноцентромеры $Centromere^*$ определим $minCount(Monomers) = min_{M \in Monomers} count(M, Centromere^*)$. $HORmon$ использует множетсво мономеров $MonomersNew^+$ для того, что бы получить моноцентромеру $Centromere^*$, далее генерируется граф де Брюина $DB(Centromere^*, 2)$ и удаляет все рёбра кратности меньше $min(MinEdgeMultiplicity, CountFraction \cdot Count(MonomerNew^+))$ с значениями по умолчанию $MinEdgeMultiplicity = 100, minCountFraction = 0.9$. На рисунке 6 изображены все сгенерированные мономерные графы.

Мономерные графы для 10 центромер(3, 11, 14, 16, 17, 19, 20, 21, 22 и X) образуют цикл и для них получение канонического HOR тривиально. Мономерный граф для центромеры 17 состоит из двух циклов: высокопокрытый цикл соответствует массиву S3C17H1L(D17Z1), а низкопокрытый цикл соответствует сестринскому HOR S3C17H1-B(D17Z1-B).

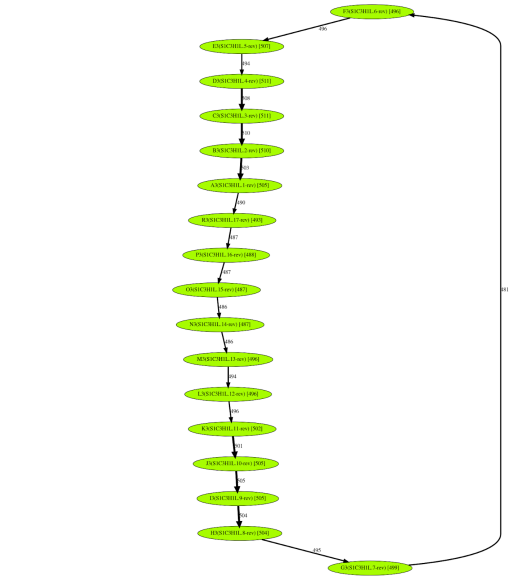
Остальные мономерные графы(хоть и неявно) тоже содержат информацию о каноническом HOR. Для получения канонического HOR, $HORmon$ строит упрощенный мономерный граф.


B1/5/19(S1C1/5/19H1L.2) [1277]
1236
C1/5/19(S1C1/5/19H1L.3) [1302]
1279
N1/5/19+D1/5/16/19+P1(S1C1/5/19H1L.4) [5637]
5557
4355
U1+L1+E1/5/16/19+F1/5/16/19(S1C1/5/19H1L.5) [5614]
1231
I1/19(S1C1/5/19H1L.6) [1267]
1234
T1(S1C1/5/19H1L.1) [1252]
1248

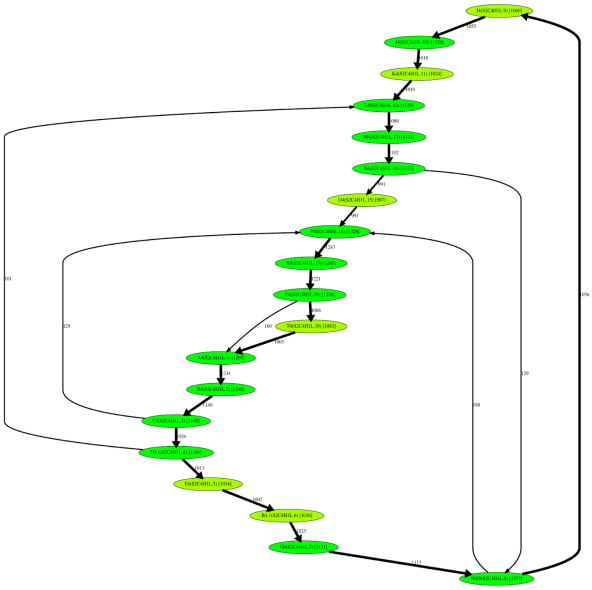
1


A2(S2C2H1L.1) [2904]
2765
247
B2/20(S2C2H1L.2) [3641]
3384
870
129
2651
C2/20(S2C2H1L.3) [3967]
3096
579
D2(S2C2H1L.4) [3261]

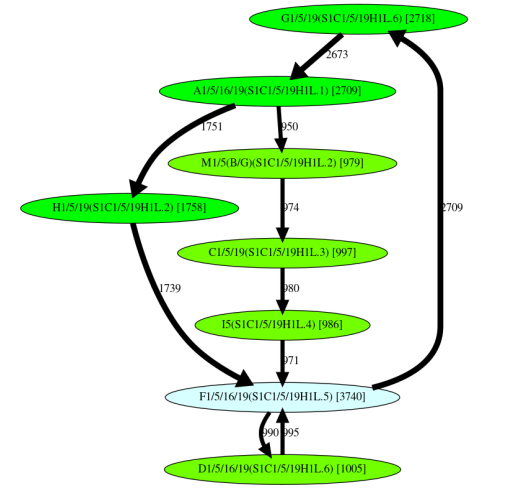
2



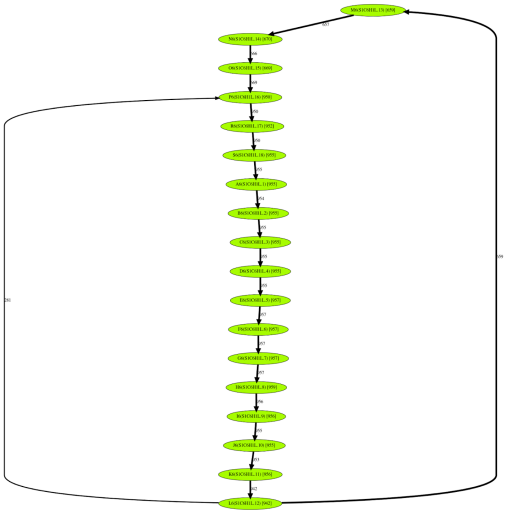
3



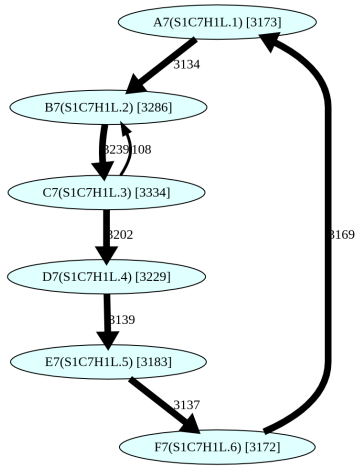
4


G1/5/19(S1C1/5/19H1L.6) [2718]
2673
A1/5/16/19(S1C1/5/19H1L.1) [2709]
1751
950
M1/5(B/G)(S1C1/5/19H1L.2) [979]
H1/5/19(S1C1/5/19H1L.2) [1758]
974
2709
C1/5/19(S1C1/5/19H1L.3) [997]
1739
980
I5(S1C1/5/19H1L.4) [986]
971
F1/5/16/19(S1C1/5/19H1L.5) [3740]
990
995
D1/5/16/19(S1C1/5/19H1L.6) [1005]

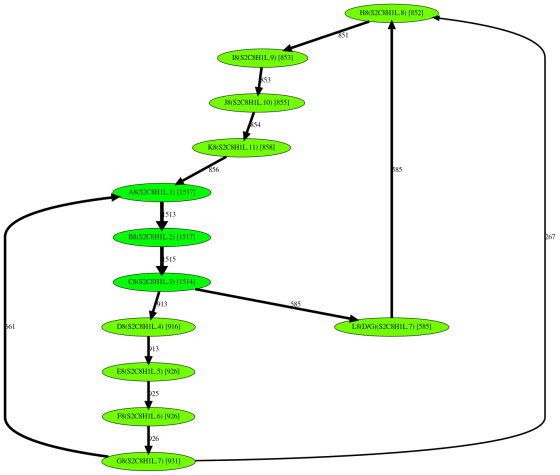
5



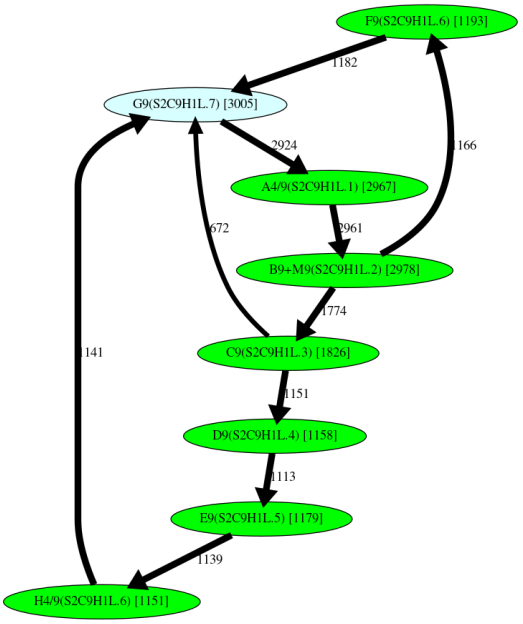
6


A7(S1C7H1L.1) [3173]
3134
B7(S1C7H1L.2) [3286]
3239
108
C7(S1C7H1L.3) [3334]
3202
D7(S1C7H1L.4) [3229]
3139
E7(S1C7H1L.5) [3183]
3137
F7(S1C7H1L.6) [3172]
3169

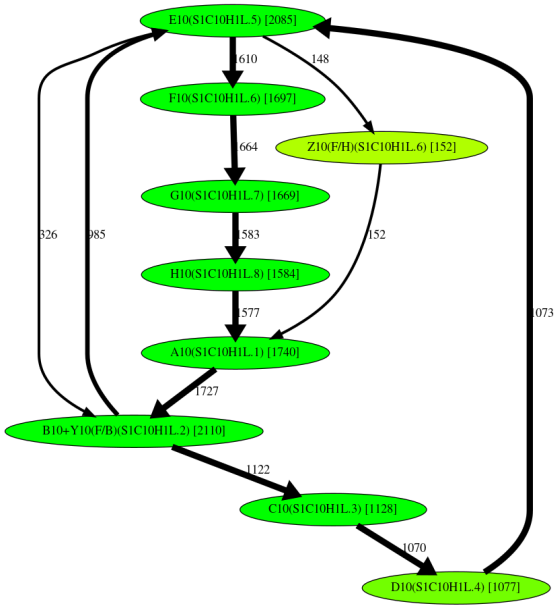
7


H8(S2C8H1L.8) [852]
851
I8(S2C8H1L.9) [853]
853
J8(S2C8H1L.10) [855]
854
K8(S2C8H1L.11) [858]
856
A9(S2C9H1L.1) [513]
513
B9(S2C9H1L.2) [517]
515
C9(S2C9H1L.3) [514]
913
585
D9(S2C9H1L.4) [916]
913
E9(S2C9H1L.5) [926]
925
F9(S2C9H1L.6) [926]
926
G9(S2C9H1L.7) [931]
L8/DG(S2C8H1L.7) [585]
585
267
961

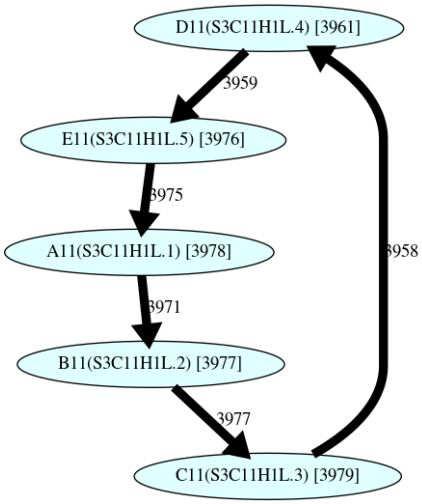
8


F9(S2C9H1L.6) [1193]
1182
G9(S2C9H1L.7) [3005]
2924
1166
A4/9(S2C9H1L.1) [2967]
672
2961
B9+M9(S2C9H1L.2) [2978]
1774
141
C9(S2C9H1L.3) [1826]
1151
D9(S2C9H1L.4) [1158]
1113
E9(S2C9H1L.5) [1179]
1139
H4/9(S2C9H1L.6) [1151]

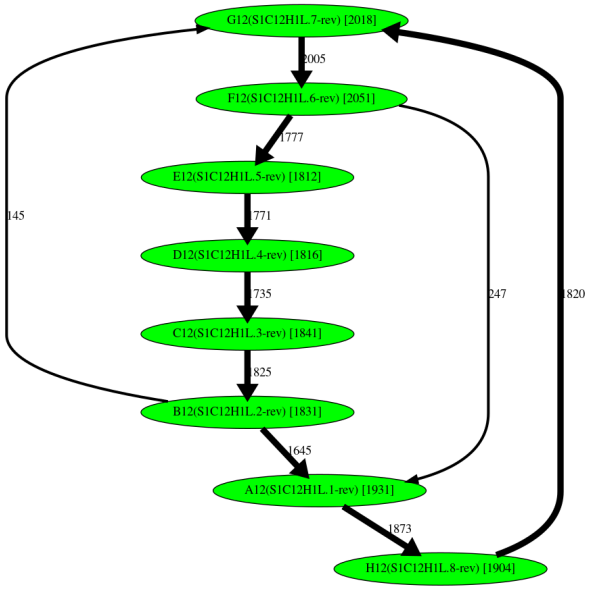
9


E10(S1C10H1L.5) [2085]
1610
148
F10(S1C10H1L.6) [1697]
664
Z10(F/H)(S1C10H1L.6) [152]
G10(S1C10H1L.7) [1669]
326
985
583
152
H10(S1C10H1L.8) [1584]
577
1073
A10(S1C10H1L.1) [1740]
1727
B10+Y10(F/B)(S1C10H1L.2) [2110]
1122
C10(S1C10H1L.3) [1128]
1070
D10(S1C10H1L.4) [1077]

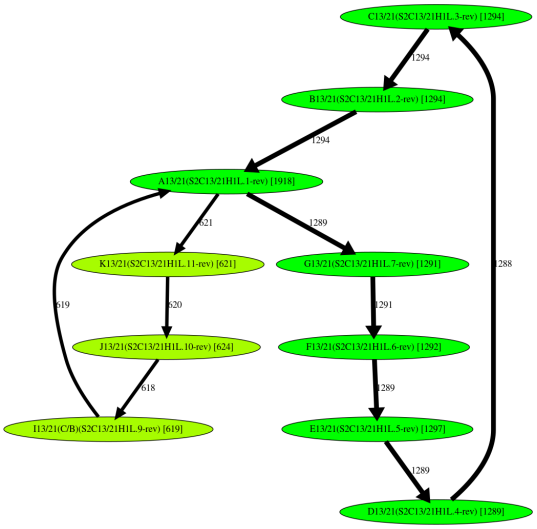
10


D11(S3C11H1L.4) [3961]
3959
E11(S3C11H1L.5) [3976]
3975
A11(S3C11H1L.1) [3978]
3958
3971
B11(S3C11H1L.2) [3977]
3977
C11(S3C11H1L.3) [3979]

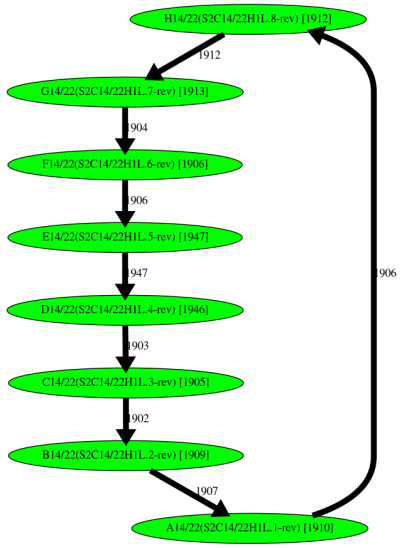
11


G12(S1C12H1L.7-rev) [2018]
2005
F12(S1C12H1L.6-rev) [2051]
1777
E12(S1C12H1L.5-rev) [1812]
145
771
D12(S1C12H1L.4-rev) [1816]
247
820
735
C12(S1C12H1L.3-rev) [1841]
825
B12(S1C12H1L.2-rev) [1831]
1645
A12(S1C12H1L.1-rev) [1931]
1873
H12(S1C12H1L.8-rev) [1904]

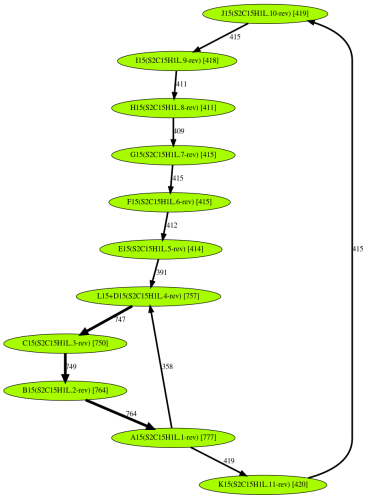
12



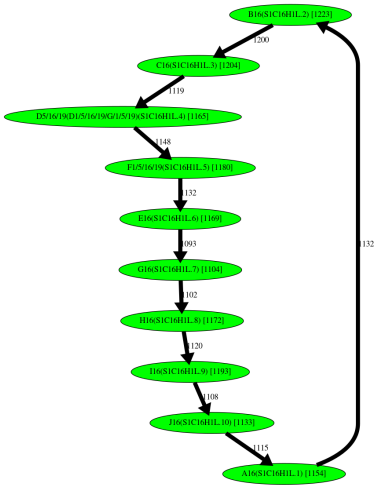
13



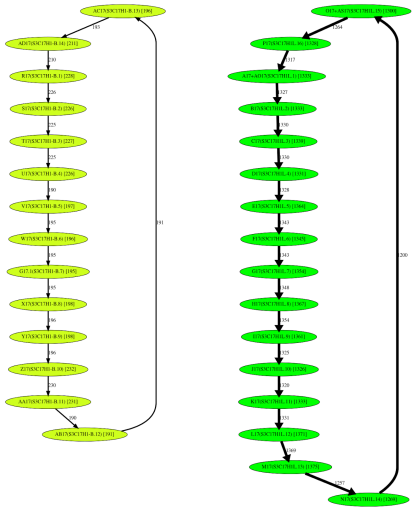
14



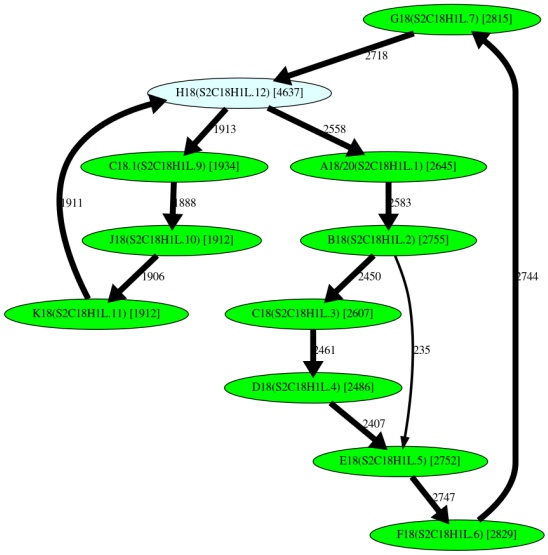
15



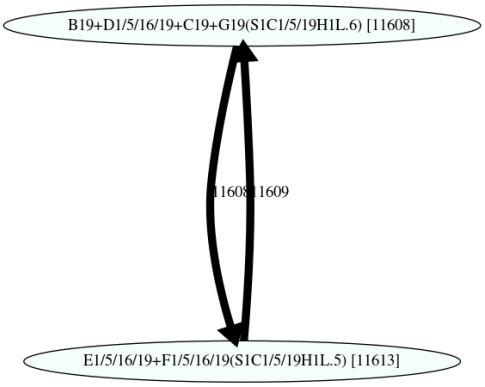
16



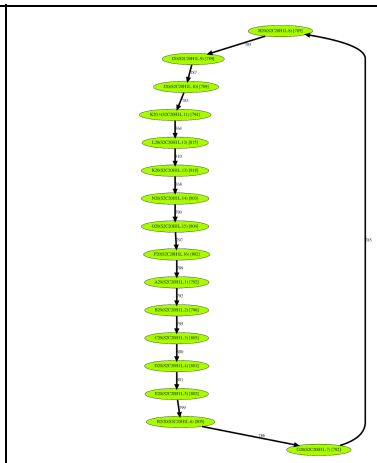
17



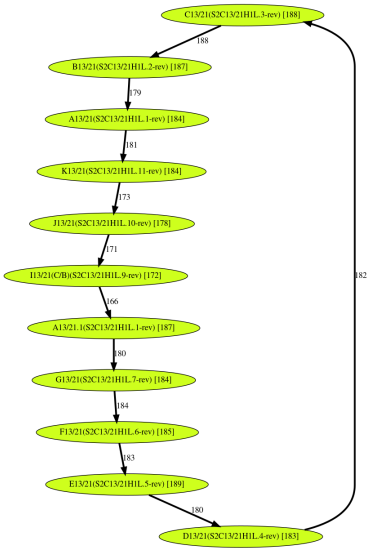
18


B19+D1/5/16/19+C19+G19(S1C1/5/19H1L.6) [11608]
11608
11609
E1/5/16/19+F1/5/16/19(S1C1/5/19H1L.5) [11613]

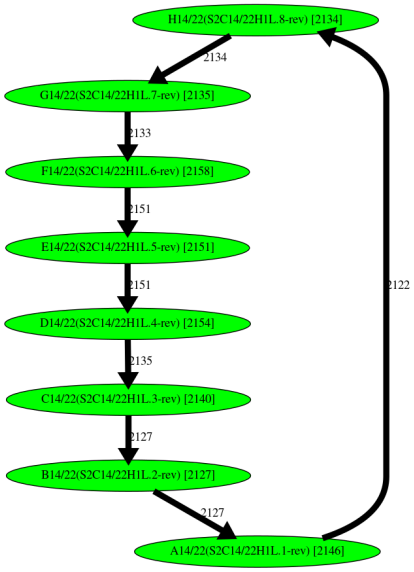
19



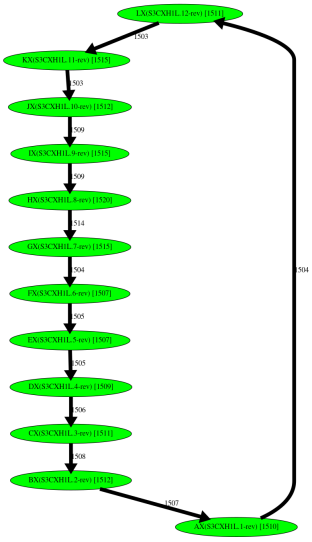
20


C13/21(S2C13/21H1L.3-rev) [188]
188
B13/21(S2C13/21H1L.2-rev) [187]
179
A13/21(S2C13/21H1L.1-rev) [184]
181
K13/21(S2C13/21H1L.11-rev) [184]
173
J13/21(S2C13/21H1L.10-rev) [178]
171
I13/21(C/I)(S2C13/21H1L.9-rev) [172]
166
A13/21.1(S2C13/21H1L.1-rev) [187]
180
G13/21(S2C13/21H1L.7-rev) [184]
184
F13/21(S2C13/21H1L.6-rev) [185]
183
E13/21(S2C13/21H1L.5-rev) [189]
180
D13/21(S2C13/21H1L.4-rev) [183]
182

21


H14/22(S2C14/22H1L.8-rev) [2134]
2134
G14/22(S2C14/22H1L.7-rev) [2135]
2133
F14/22(S2C14/22H1L.6-rev) [2158]
2151
E14/22(S2C14/22H1L.5-rev) [2151]
2151
D14/22(S2C14/22H1L.4-rev) [2154]
2135
C14/22(S2C14/22H1L.3-rev) [2140]
2127
B14/22(S2C14/22H1L.2-rev) [2127]
2127
A14/22(S2C14/22H1L.1-rev) [2146]
2122

22


LX(S3CXH1L.12-rev) [1511]
1503
KX(S3CXH1L.11-rev) [1515]
1503
JX(S3CXH1L.10-rev) [1512]
1509
IX(S3CXH1L.9-rev) [1515]
1509
HX(S3CXH1L.8-rev) [1520]
1514
GX(S3CXH1L.7-rev) [1515]
1504
FX(S3CXH1L.6-rev) [1507]
1505
EX(S3CXH1L.5-rev) [1507]
1505
DX(S3CXH1L.4-rev) [1509]
1506
CX(S3CXH1L.3-rev) [1511]
1508
BX(S3CXH1L.2-rev) [1512]
1507
AX(S3CXH1L.1-rev) [1510]
1504

X

Рис. 6: **Мономерные графы построенные для всех человеческих центромер с помощью** $HORmon$. На каждой вершине указаны имена мономеров(и имена, которые соответствуют именам в T2T консорциуме согласно [7]) и их количество в моноцентромере(в скобочках). Мономеры полученные путем склейки двух мономеров помечены как сумма двух мономеров. Например, вершина с названием "N1/19+D1/5/16/19+P1"в мономерном графе для 1 центромере получена в результате склейки мономеров $N1/5/19$, $D1/5/16/19$ и $P1$. Мономеры полученные в результате расклейки мономера помечены как оригинальное имя и ещё одно число через точку, обозначающие порядковый номер расклеенного мономера. Например, мономера $B4$ и $B4.1$ в мономерном графе для 4 центромеры – это мономеры полученные в результате расклейки мономера $B4$. На ребрах написан их вес. Толщина ребра(цвет вершины) соответствует весу ребра(количеству мономеров).

3.6. Упрощенный мономерные графы

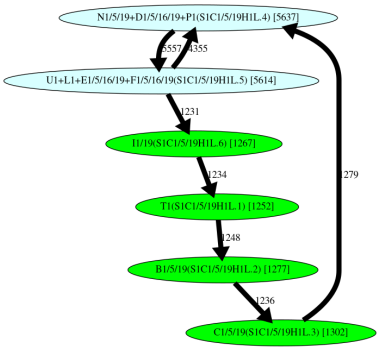
Алгоритм построения

Для заданного мономерного графа, $HORmon$ строит полный двудольный граф, где каждая доля состоит из всех вершин(мономеров) из мономерного графа. Мономер M' из первой доли соединяется с вершиной M'' из второй доли и вес ребра равен весу ребра (M', M'') в мономерном графе. После этого $HORmon$ решает задачу о назначение и находит паросочетание максимального веса ([1]). Ребра в двудольном графе соответствуют ребрам в мономерном графе и максимальное паросочетание образует в мономерном графе непересекающиеся пути и циклы. Ребро в мономерном графе классифицирается как удаляемое, если оно является хордой в одном из этих циклов или путей(определим хорду для пути, как ребро, которое соединяет две внутренние вершины). Мономерный граф, в котором удалили все удаляемые ребра – это *упрощенный мономерный граф*

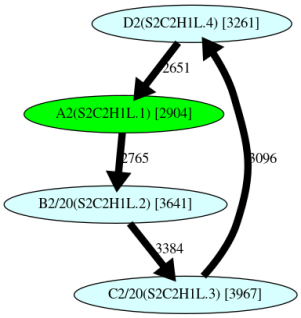
Упрощенные графы для человеческих центромер

На рисунке 7 изображены упрощенные мономерные графы. В случае если упрощенный граф состоит из цикла, то из них можно легко вывести канонический HOR. Из упрощенных мономерных графов канонический HOR выводится для всех центромер, кроме 1, 5, 8, 9, 10, 13 и 18. Поскольку в первой центромере есть единственный гамильтонов цикл, я его классифицирую как HOR. Центромера 1 хорошо иллюстрирует, что помимо выявленных HORов(для большенства центромер), упрощенные мономерные графы содержат дополнительную эволюционную информацию, которую не содержат HORы. Например, в мономерном графе для 1 центромеры есть два антипараллельных ребра большой кратности, которые не отражены в HOR для первой центромере. На самом деле, изначально как канонический HOR в первой центромере определялся именно этот димер([3])

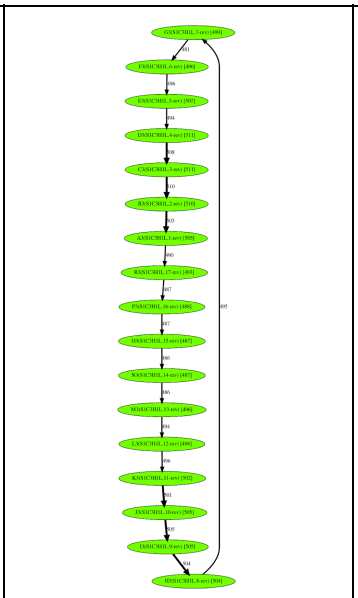
В оставшихся 6 центромер(5, 8, 9, 10, 13 и 18) нет гамилтонова цикла и эти центромеры я рассмотрю далее отдельно.



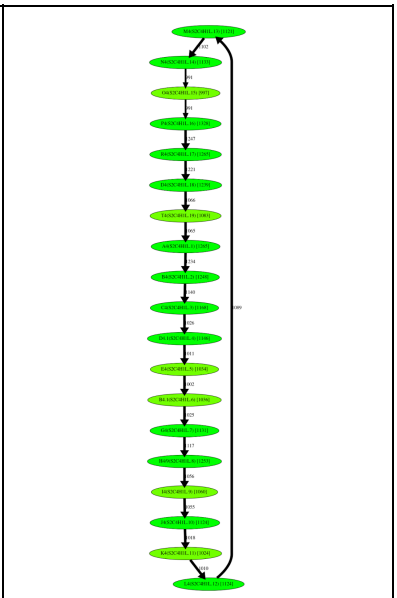
1



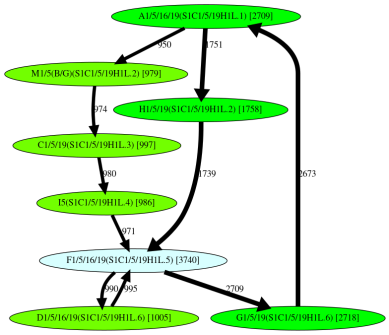
2



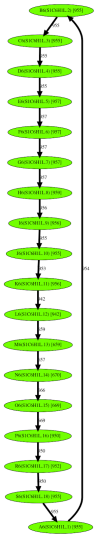
3



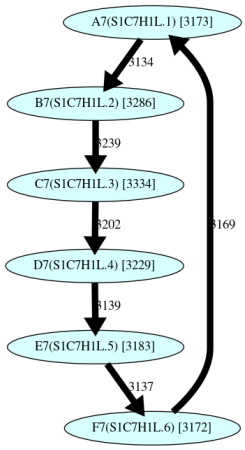
4



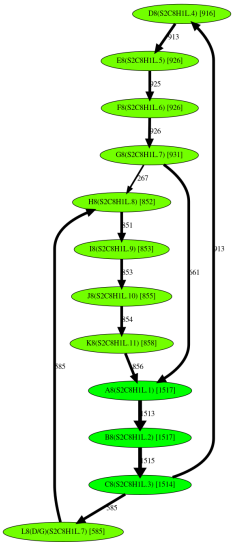
5



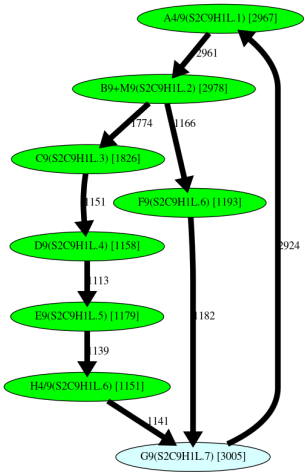
6



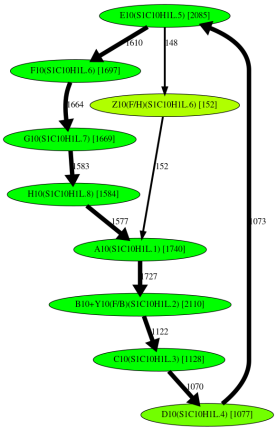
7



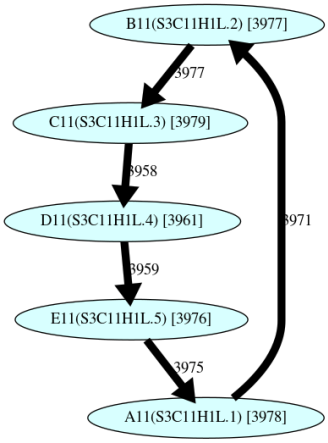
8



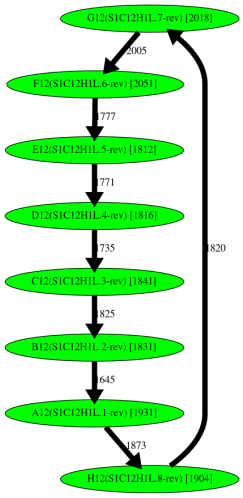
9



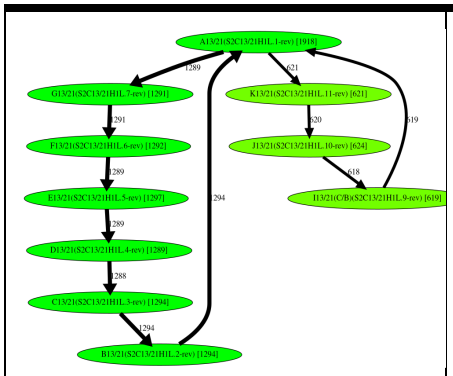
10



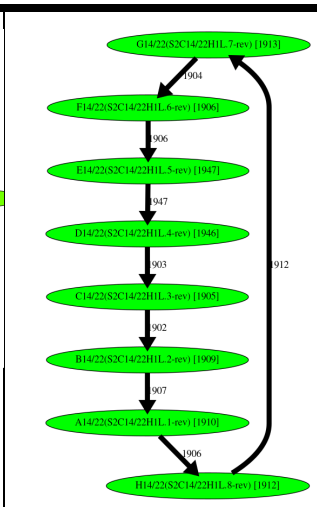
11



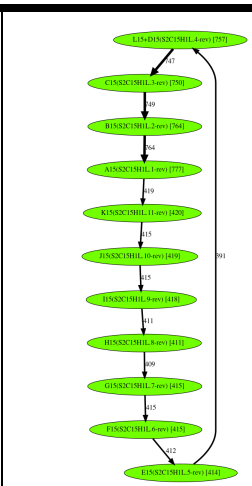
12



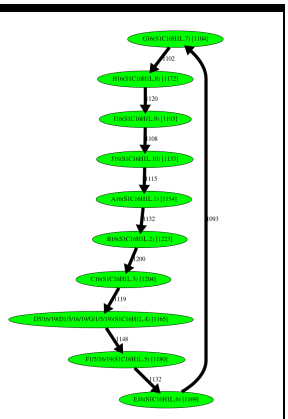
13



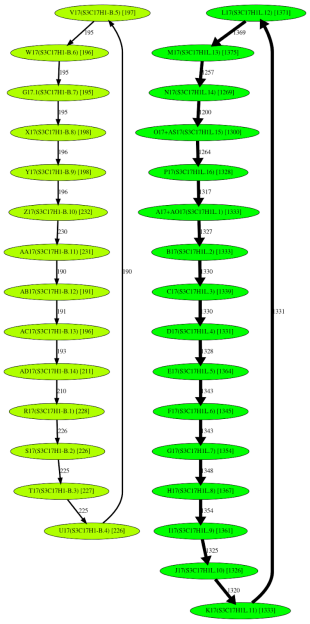
14



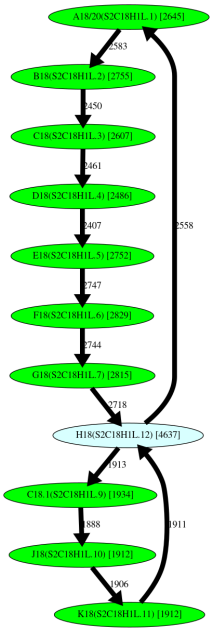
15



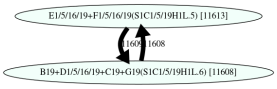
16



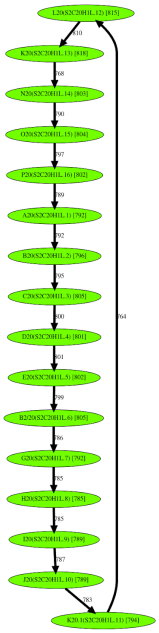
17



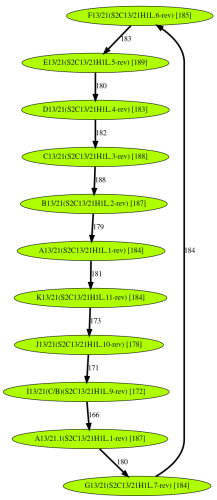
18



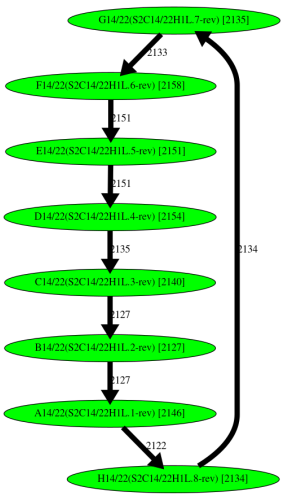
19



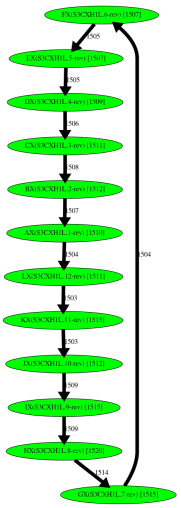
20



21



22



X

Рис. 7: **Упрощённые мономерные графы для всех человеческих центромер(кроме Y)**. На каждой вершине указано имя мономера и его количество в моноцентромере(в скобочках). На ребрах указан вес. Толщина ребра(цвет вершины) соответствует их весу(количеству мономеров). Имена мономеров соответствует конвенции из [4]. Y центромера не рассматривается, поскольку для неё ещё нет сборки.

4. Модель для структуры центромеры

4.1. Ограничения CE постулата

На картинке 8 показаны две игрушечные моноцентромеры, которые иллюстрирует ограничения текущей концепции HOR. Не смотря на то, что для каждой центромеры можно придумать разумный эволюционный сценарий остается совершенно не ясно, какими должны быть HORы в данном случае согласно CE постулату.

Действительно, для данных моноцентромер мы получаем одинаковые мономерные графы(два цикла AB и BA соединенные общей вершиной B), отражают очень разные эволюционные сценарии. Первую моноцентромеру можно представить как два цикла (BA) и (BC), а вторая может быть описана единственным циклом $BABC$ в мономерном графе.

Концепция HOR не позволяет достаточно хорошо продемонстрировать различия между этими двумя игрушечными центромерами показанными на рисунке **??**. Поскольку необходимо, что бы каждый мономер встречался в HOR ровно один раз, единственный кандидат для HOR это ABC(и это не является адекватным отражением центромеры). Хотя этот пример можно считать искусственным(он противоречит CE постулату о том, что канонический HOR должен существовать для любого альфа-сателитного массива), любой алгоритм аннотации центромер должен адекватно обрабатывать такие случаи, даже если они редко появляются в человеческих центромерах. Ниже будет показано, что в центромерах 13 и 18 ситуация будет очень похожа на этот игрушечный пример. Далее будет описана концепция моноран графа, которая дает более информативное представление об архитектуре центромеры.

4.2. Разделение не разделяемых мономеров для центромер 13 и 18

Хотя упрощённый мономерный граф для центромер 13 и 18 представляют из себя два цикла соединенные вершиной(а я следую определе-

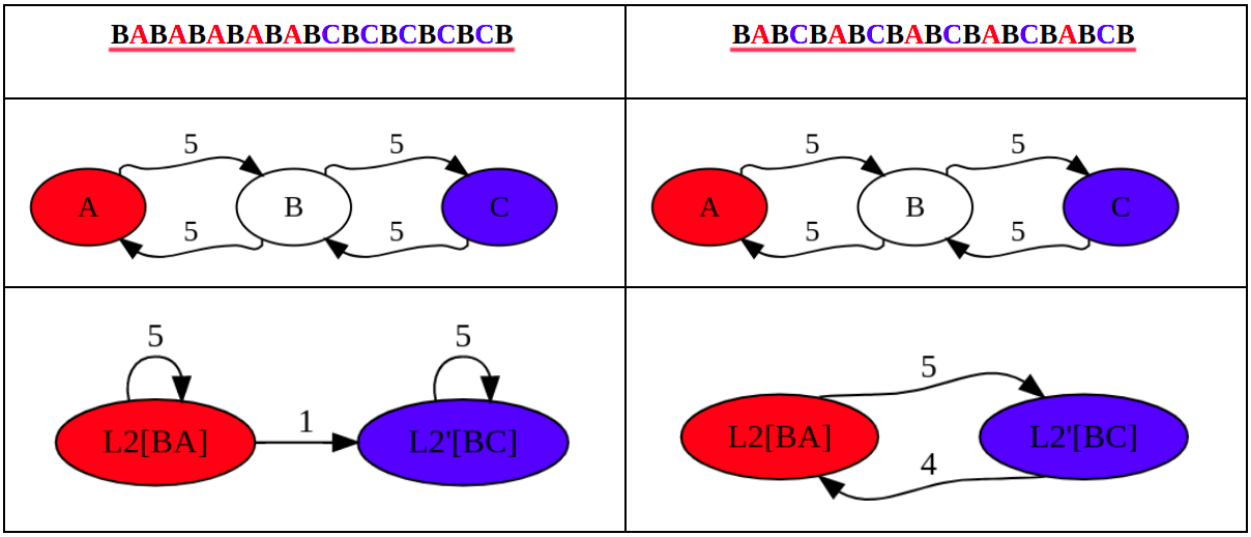


Рис. 8: "Моноцентромеры"BABABABABABCBCBCBCBCB и BABCBABCBABCBABCBABCB(верхняя строка), де Брюин графы $DB(Centromere^*, 2)$, который соответствует мономерному графу(вторая строка). Хотя мономерные графы одинаковые, моноран графы(будут описаны ниже) разные(нижняя строка)

нию HOR где каждая вершина должна встречаться только один раз), можно преобразовать в один цикл разделив эту вершину на две части(рисунок 9). Поскольку так или иначе не разделяемая вершина, её разделение будет конфликтовать с каким-то из частых путей или разделение этой вершиной может приводить к паре очень близких мономеров. И действительно, при расклейки мономеров для 13(18) центромеры я получила мономеры которые отличаются всего на 3(1) нуклеотида. Я в любом случае сделала эту операцию, что бы быть консистентной с CE постулатом.

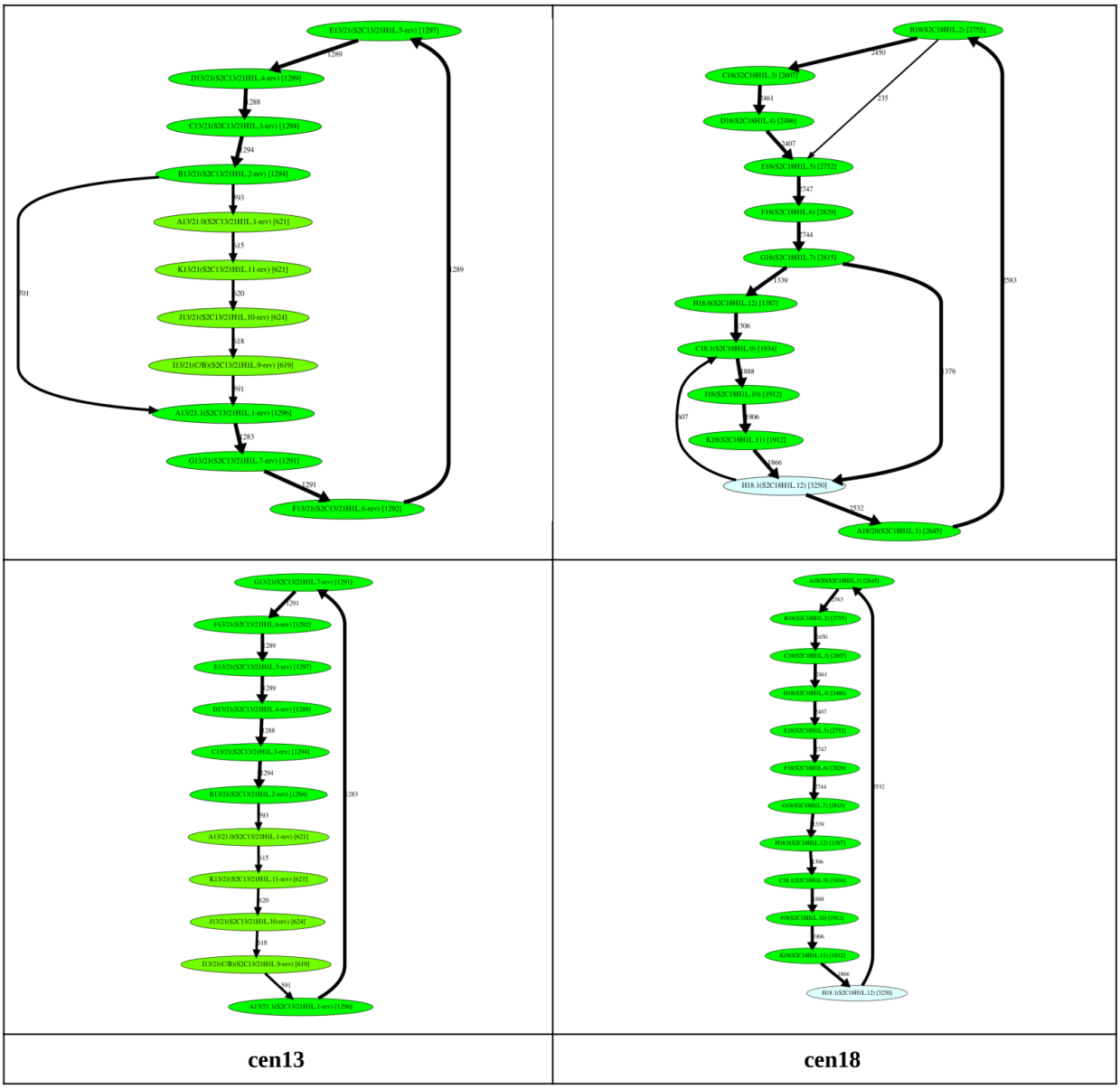


Рис. 9: **Разделение не разделяемых мономеров для упрощённых мономерных графов для центромеры 13(слева) и центромеры 18(справа).** Разделение не разделяемого мономера в 13 центромере привело к двум мономерам которые отличаются всего в трёх нуклеотидах и мономер граф для центромеры стал иметь вид цикла с одной хордой(сверху слева). В результате упрощённый мономерный граф стал иметь вид цикла который является каноническим 11-мономерным HOR в 13 центромере. Разделение не разделяемого мономера в 18 центромере привело к двум мономерам, которые отличаются всего в одном нуклеотиде и привело к образованию цикла с тремя хордами(справа сверху). Итоговый упрощенный граф(справа снизу) – это 12 мономерный канонический HOR для 18 центромеры.

4.3. Гибридные мономеры усложняют выделения HOR для центромер 5, 8 и 10

Для поиска гибридов в множестве мономеров $MonomersNew^{+}$ использовался метод описанный в разделе 1.6. В результате этого анализа было выявлено только 4 частых гибрида: D5 и H5 в центромере 5, L8 в центромере 8 и Z10 в центромере 10. Далее будет описана операция гибридной декомпозиции в мономерном графе для выявления HORов для 5, 8 и 10 центромер.

Гибридная декомпозиция для центромеры 10. Мономер $Z10$ – это гибрид мономеров F10 и H10, полученный в результате конкатинации первых 88 нуклеотидов F10 и последних 82 нуклеотидов в H10(обозначим как F10(88)+H10(82)). Эта конкатинация отличается от мономера Z10 только в 3 нуклеотидах. На рисунке 10 мономеры F10, H10 и Z10 показаны синими, красным и сине-красным цветом соответственно. На картинке 10 показана гибридная декомпозиция из которой удалили вершину $Z10$ и вместо неё добавили гибридное ребро из левой половины F10 в правую половину H10. Добавление гибридного ребра является важным шагом, поскольку иначе мономерный граф не будет адекватным представлением моноцентромеры. Поскольку теперь есть один гамильтонов цикл(рисунок 10 снизу), я его классифицирую как HOR.

Гибридная декомпозиция для центромеры 8. Аналогично, L8 является гибридным мономером $D8(60) + G8(111)$ и такая конкатинация отличается от L8 только на 2 нуклеотида. Рисунок 11 иллюстрирует гибридную декомпозицию для мономера L8, что приводит к мономерному графу с единственным гамильтоновым циклом и гибридной хордой.

Гибридная декомпозиция для центромеры 5. $D5$ – это гибридный мономер $G5(50) + I5(120)$ мономера $G5$ и $I5$(конкатинация отличается от D5 на 5 нуклеотидов), H5 – это гибрид $M5(92) + I5(78)$(эта конкатинация отличается от H5 на 6 нуклеотидов). На рисунке 12 показана декомпозиция и гибридов для D5 и H5, в которой эти мономеры заменены на гибридное ребро в мономерном графе, и в результате был получен граф с единственным гамильтоновым циклом, который был

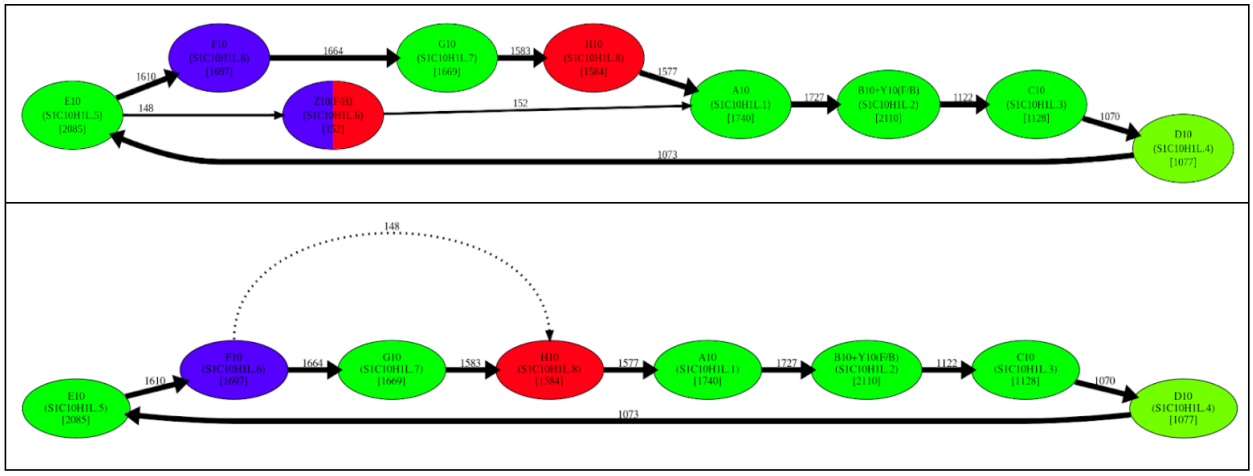


Рис. 10: **Гибридная декомпозиция мономера Z10** (Сверху) для того что бы показать, что мономер $Z10$ является гибридом мономеров F10 и H10, вершины F10, H10 и Z10 в мономерном графе нарисованы синим, красным и сине-красным цветом соответственно. (Снизу) Удаление гибридной вершины Z10 из упрощённого мономерного графа и добавление гибридного ребра(нарисован пунктиром) приводит к графу из одного цикла и одной хорды

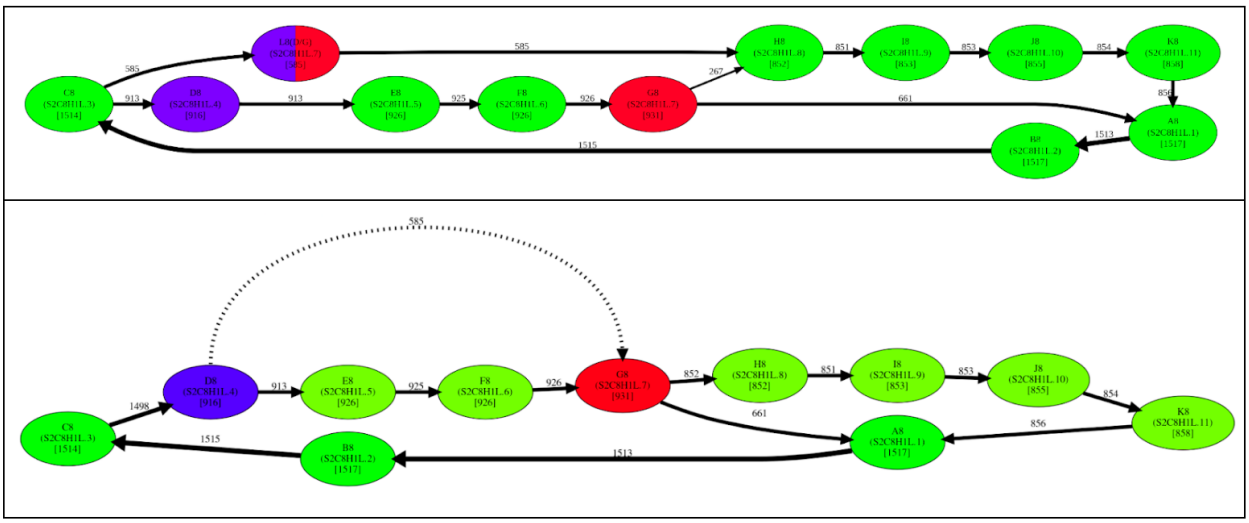


Рис. 11: **Гибридная декомпозиция мономера L8**

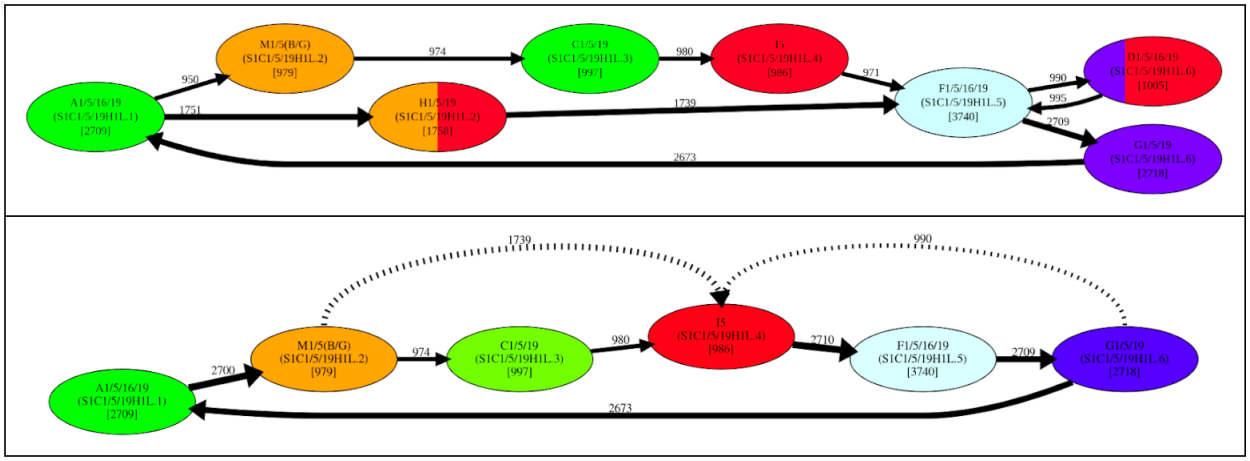


Рис. 12: **Гибридная декомпозиция мономера D5 и H5**

классифицирован как HOR.

4.4. HOR для центромеры 9

После разделения не разделяемых вершин(в центромерах 13 и 18) и гибридной декомпозиции(в центромерах 5, 8 и 10) были получены эволюционно адекватные HORы для всех центромер, кроме 9. Это центромера – это наиболее сложный случай с точки зрения CE постулата, поскольку не ясно как именно вывести HOR для данного мономерного графа. На рисунке 13 синим показан путь, который сейчас считается HORом(выведенным в ручную). Мономер F9(который не относится к HOR в центромерер 9) не является гибридным мономером и отличается от ближайшего мономера $H4/9$ в центромере 9 на 12 нуклеотидов. Сейчас не понятно, как именно автоматически выделять канонический HOR для 9 центромеры.

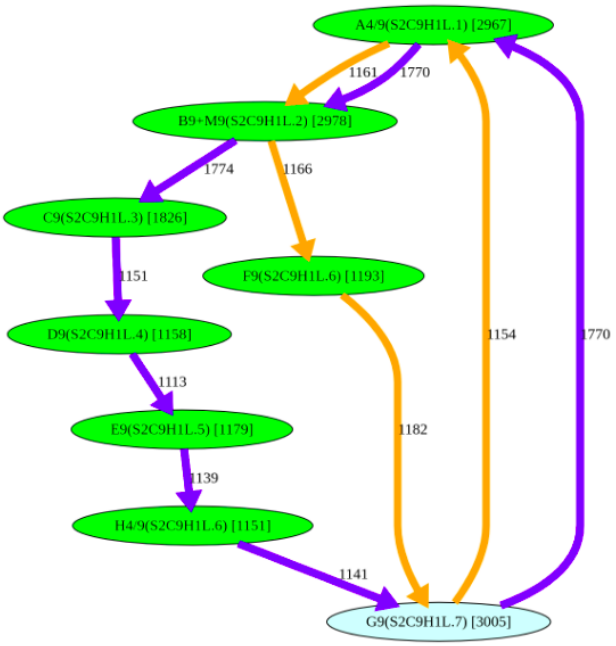


Рис. 13: **HOR для центромеры 9 показан как путь обходящий 7 из 8 мономеров в мономерном графе** В ручную выделенный HOR в центромере 9 показан как синий цикл, это противоречит с CE постулатом, потому что частый жёлтый путь содержит мономер, который не принадлежит синему циклу

4.5. Моноциклы

Проход моноцентромеры от начала до конца определяются обходом мономерного графа. Поскольку *HORmon* удаляет ребра малого веса из мономерного графа обход не обязательно будет соответствовать проходу по одному ребру в мономерном графе, а скорее проходу по пути ребер. Говорим, что последовательность обходит цикл $e_1, \cdots, e_n$ в мономерном графе, если после последовательного прохода по $e_1, \cdots, e_n$ далее опять происходит проход по e_1.

Определим *моноцикл* как цикл который обходит моноцентромера хотя бы $minTraversals$ раз(по умолчанию 100). Заметим, что моноцикл может иметь повторяющиеся вершины(но не ребра). Если мономерный граф представляет из себя единственный цикл(как например для случая с 3, 11, 14, 16, 19, 20, 21, 22 и X, рисунок 3) или два изолированных цикла(центромера 17), тогда концепты HOR и моноцикла совпадают. Однако, мономерный графы для других центромер совершенно не от-

ражают как часто происходит обход каждого из циклов. Например, в игрушечном примере с моноцентромерой(рисунок 8) мономерные графы одинаковые(два цикла AB и BC соединенные общей вершиной B), однако они отражают очень разные эволюционные истории. Первый мономерный граф состоит из двух моноциклов: первый состоит из ребер AB и BA, а второй из ребер CB и BC. Второй мономерный граф состоит из единственного моноцикла образованного ребрами AB, BC, CB и BA. Изображение архитектуры центромер в виде моноцентромер более информативная, чем представление в виде HORов(которая больше направлена на порядок мономеров в предка, чем изображению текущей архитектуры). Для анализа моноциклов в следующем разделе я определю понятие моноран графа.

4.6. Моноран графы

Определим моноран как не ветвящийся путь в мономерном графе. Вес монорана определяется как минимальный вес ребер из которых состоит моноран, а длина монорана – это количество ребер в моноране. Например, центромера 15 состоит из 3 линейных мономеров длины 1, 3 и 8, которые я обозначаю как L1, L3 и L8 соответственно. Центромера 13 состоит из 2 циклических мономеров длины 4 и 7, которые я обозначаю как C4 и C7 и центромера 8 состоит из 6 линейных мономеров длины 1, 1, 2, 2, 4 и 4.(Рисунок 14)

Поскольку мономеры представляют из себя более компактный язык для описания архитектуры центромера, чем мономеры, центромера была переписана в алфавите мономеров, в которой удалены символы, которые не принадлежат монранам и был повторен моноран граф как де Брюин граф над итоговой строкой для k=2. Эту процедуро итеративно и получать всё более топологически простые моноран графы. Рисунок **??**(внизу) показывает, что моноран граф может лучше отражать архитекутру центромер, чем мономерный граф.

Рисунок 14 отражает моноран графы для центромеры 15, 13 и 8. Моноран граф для центромеры 15 состоит из трёх вершин(мономеров):

L1(вес 358), L3(вес 747) и L8(вес 391). Здесь мономер граф отражает моноциклы L3L1 и L3L8, но не показывает, как часто в центромере 15 происходит переход между этими двумя моноциклами. Что бы построить ещё более детальное представление центромеры 15 я взяла не ветвящиеся пути в мономер графе (L3L1 и L3L8) и построила мономер граф на этих ребрах(Рисунок 9 снизу) из чего видно, что моноцикл L3L8, который соответствует каноническому HOR встречается в центромере 15 как тандемный повтор всего 86 раз. Чаще всего за этим моноциклом следует моноцикл L3L1(302 раза). Такой итеративный мономер граф даёт гораздо больше информации об архитектуре центромеры 15, чем её HOR и при этом HOR – это один из моноциклов.

Мономер граф в центромерер 13(рисунок 14) состоит из 2 вершин(мономеров): C4(вес 618) и C7(вес 1288). Что бы построить более детальное представление центромеры 13 были рассмотрены не ветвящиеся пути в мономер графе(C4C7 и C7) и построен мономер граф следующего порядка на двух вершинах(рисунок 14 внизу).

Мономерный граф для 8 центромеры(так же как и для центромер 2, 4, 5, 7, 9, 10, 12, 14, 15 и 18) более сложные и из них автоматически не выводятся моноциклы даже после многократного повторения процедуры. Ниже описана операция эквивалентной трансформации мономеров, которая позволяет выделить моноциклы и их вес.

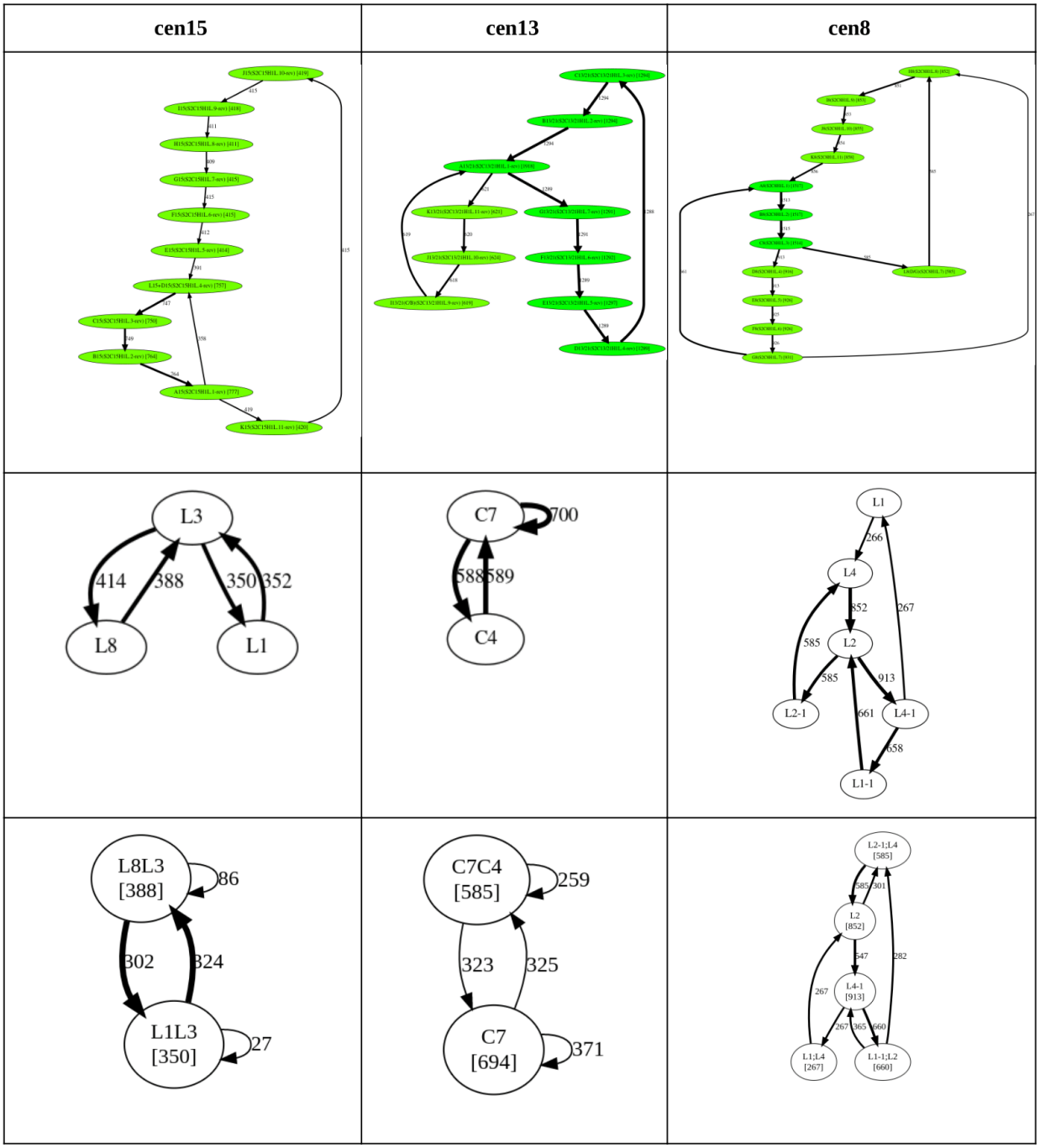


Рис. 14: **Преобразование мономерных графов в моноран графы для центомер 15, 13 и 8** В первой, второй и третьей строки показывают мономерные графы, моноран графы и итеративный моноран графы. (Слева) мономерный граф для 15 центромеры состоит из 3 линейных монорана(длины 1, 3 или 8) отображены вершинами L1, L3 и L8 в моноран графе. Моноран граф отражает кратность циклов L3L1 и L3L8. Итеративный монран граф показывает ещё более точное представление архитектуры 15 центромеры. (Средний столбец) мономерный граф для 13 центромеры, который содержит два циклических монорана C4 и C7. (Справа) мономерный граф для 8 центромеры сосотоит из 8 линейных монрана(длины 4,4,2,2,1 и 1) соответствуют вершинам L4, L4-1, L2, L2-1, L1 и L1-1 в моноран графе

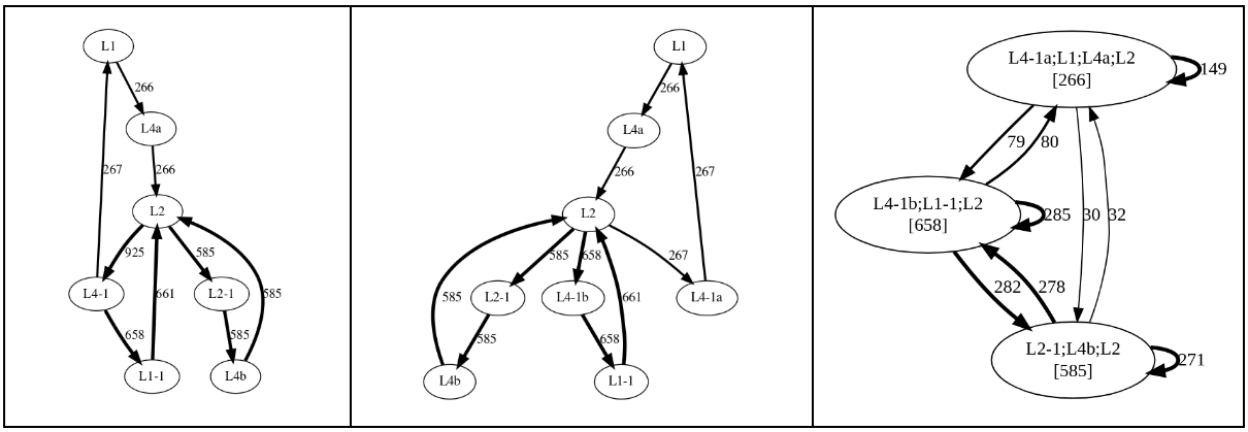
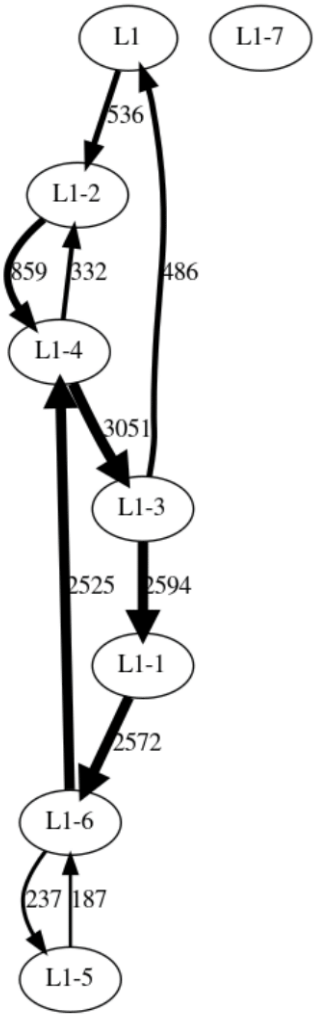
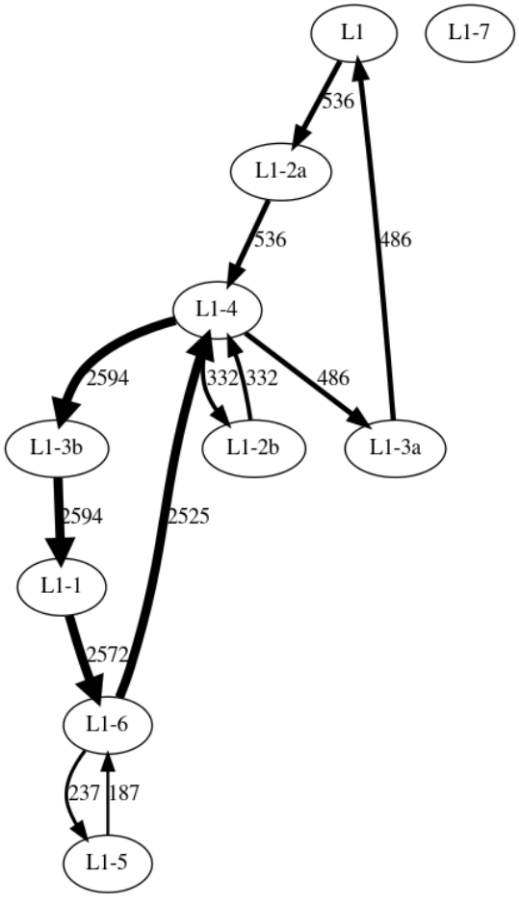
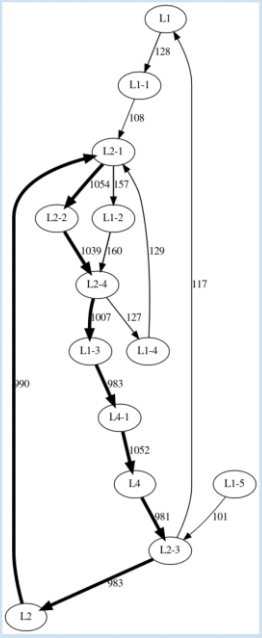
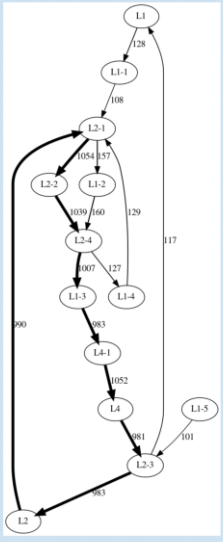


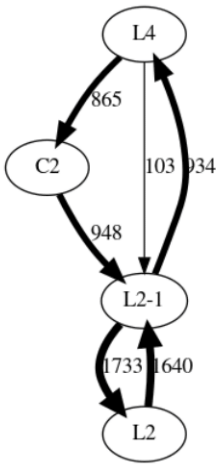
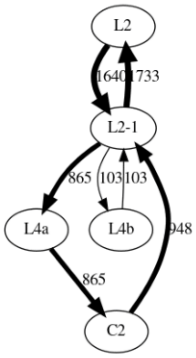
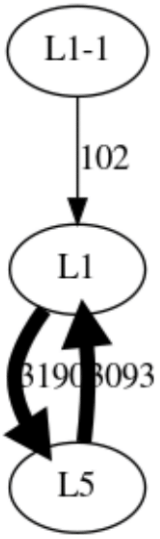
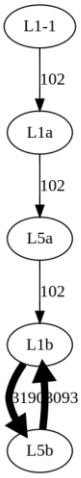
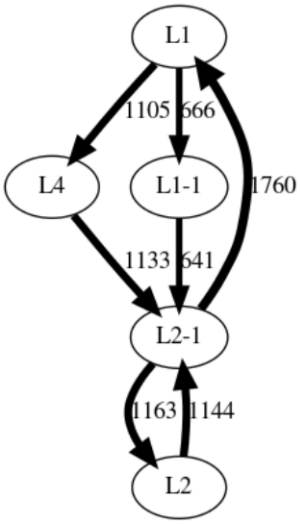
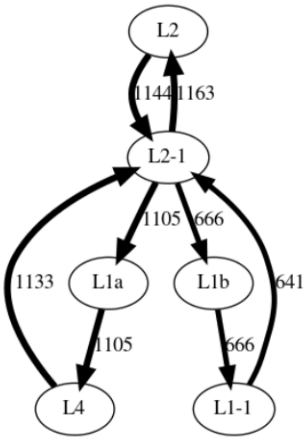
Рис. 15: **Эквивалентное преобразование мономоран графов для 8 центромеры** Мономоран графы для 8 центромеры после разделения вершины L4 на вершины L4a и L4b(слева), далее следует разделение вершины L4-1 на вершины L4-1a и L4-2b(по середине). Граф справа – это мономоран граф второго порядка

4.7. Эквивалентное преобразование графов моноранов

Заметим, что вершина $L4$ в мономоран графе для 8 центромеры – это вершина с двумя входами и одним выходом и её можно однозначно разделить на вершины $L4a$ и $L4b$(Рисунок 15) в результате чего получаем более простой граф с таким же множеством моноциклов,как и в оригинальном мономоран графе. Точно так же, вершина $L4-1$ в итоговом графе это вершина с одним входом и двумя выходами и она может быть точно так же разделена на две вершины $L4-1a$ и $L4-1b$, в результате получаем граф с 8 вершинами и тремя моноциклами(L2 + L2-1 + L4b, L2 + L4-1b + L1-1 и L2 + L4-1a + L1 + L4a) с весом 585, 685 и 266 соответственно.

Я применила эквивалентное преобразование ко всем вершинам у которых есть N входов и 1 выход или 1 вход и N выходов в мономоран графе. На рисунке 16 изображены нетрививальные графы моноранов и их эквивалентные преобразования(для центромер 2, 4, 5, 7, 9, 10, 12 и 18)

2		 **Monocycles:** • L1-4 + L1-2b — multiplicity 332 • L1-4 + L1-3a + L1 + L1-2a — multiplicity 486 • L1-4 + L1-3b + L1-1 + L1-6 — multiplicity 2572 • L1-6 + L1-5 — multiplicity 187
4		 **Monocycles:** • L2 + L2-1 + L2-2 + L2-4 + L1-3 + L4-1 + L4 + L2-3 + L2-3 — multiplicity 981

5		 **Monocycles:** • L2-1 + L4a + C2 — multiplicity 865 • L2-1 + L4b — multiplicity 103 • L2-1 + L2 — multiplicity 1640
7		 **Monocycles:** • L1b + L5b — multiplicity 3093
9		 **Monocycles:** • L2 + L2-1 — multiplicity 1144 • L2-1 + L1a + L4 — multiplicity 1105 • L2-1 + L1b + L1-1 — multiplicity 641

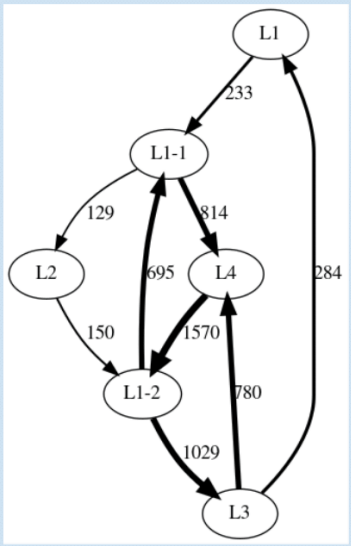
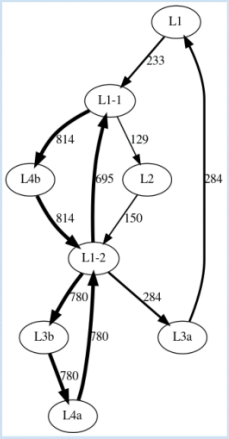
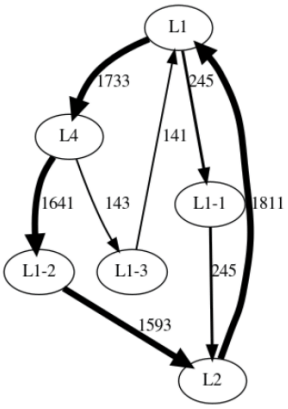
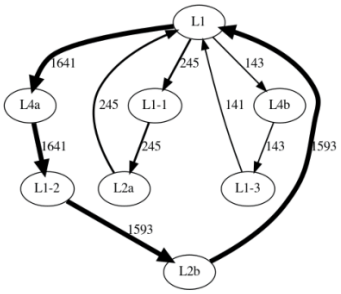
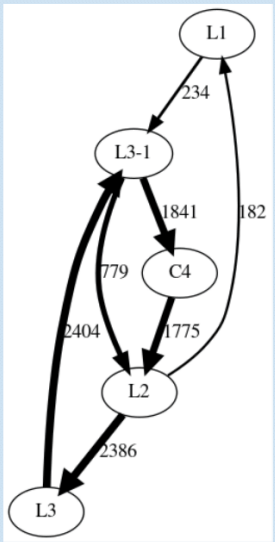
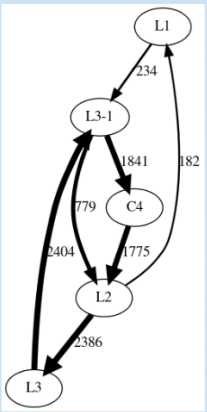
10		 **Monocycles:** • L1-2 + L1-1 + L4b — multiplicity 695 • L1-2 + L1-1 + L2 — multiplicity 129 • L1-2 + L3b + L4a — multiplicity 780 • L1 + L1-1 + L4b + L1-2 + L3a — multiplicity 233
12		 **Monocycles:** • L1 + L4b + L1-3 — multiplicity 141 • L1 + L1-1 + L2a — multiplicity 245 • L1 + L4a + L1-2 + L2b — multiplicity 1593
18		 **Monocycles:** • L3 + L3-1 + C4 + L2 — multiplicity 1775 • L3-1 + L2 + L3 — multiplicity 779 • L2 + L1 + L3-1 + C4 — multiplicity 182 • L2 + L1 + L3-1 — multiplicity 182

Рис. 16: **Моноран графы и моноран графы после эквивалентной трансформации для центромер 2, 4, 5, 7, 9, 10, 12, 15 и 18**

Заключение

В рамках данной работы был разработан инструмент для выделения мономеров и канонических HOR. Это первый инструмент, который выделяет мономеры согласовано с постулатом о центромерной эволюции.

Была поставлена и решена задача выделения мономеров, как задача кластеризации строк. Для того, что получить мономеры для всего генома, для решения применялся набор эвристик.

Я произвела сравнение сгенерированных мономеров и мономеров, которые используются в T2T консорциуме. Я оценивала мономеры по метрикам среднего квадратичного отклонения и Дэвид-Болдуин индекса. Дэвид-Болдуин индекс оказался примерно одинаковым для первого и второго случае, а среднее квадратичное отклонение у выделенных мономеров получилось лучше. Что говорит что сгенерированные мономеры лучше, с точки зрения кластеризации строк.

При этом оказалось, что не смотря на то, что решение лучше с точки зрения кластеризации, оно не является адекватным с биологической точки зрения. Поскольку не согласовывается с постулатом центромерной эволюции. Поэтому после этого я провела постобработку мономеров, что бы получить адекватные, с точки зрения биологии, результаты. Почти на всех центромерах удалось вывести мономеры близкие к T2T мономерам.

Однако,остаётся несколько случаев, которых сложно обработать автоматически: центромера 9 и 18. В 9 центромере 2 мономера, которые согласно T2T мономерам, должны объединяться в один кластер находятся на слишком большом расстояние, поэтому выделяются в два разных мономера. Возможно, в этом случае, может помочь использование другой метрики для сравнения строк. Другая проблема в 18 центромере, в которой мономер должен быть разделен на два для соответствия CE постулату. Если мономер разъединить, то два новых консенсуса будут находиться на расстояние в 1 нуклеотид и более того, некоторые блоки, которые должны относиться к разным мономерам являются полностью одинаковыми. Случай в 18 центромере плохо согласуется с

постулатом о центромерной эволюции.

Кроме того, в этой работе в дополнение к концепции HOR была введена концепция моноцикла, который не опирается на постулат центромерной эволюции. Несмотря на то, что анализ мономерных графов позволяет компактно визуализировать архитектуру центромер и автоматически вывести известные HOR для «живых» центромер человека, HOR является концепцией, сильно зависящей от параметров. Хотя эта концепция внесла большой вклад в предыдущие исследования центромер и пролила свет на организацию предковых центромер, она не обязательно оптимальна для описания существующих архитектур центромер. Таким образом, в этой работе концепция HOR дополняется более информативным понятием монорангграфа и анализом моноциклов в этих графах. Поскольку до недавнего времени не существовало сборок центромер человека, было практически невозможно точно вывести монорангграфы из (коротких) ридов и выявить, что понятие HOR сильно зависит от параметров и может не подходить для аннотации более сложных центромер.

Модули разработанные в рамках этой работы стали частью инструмента для автоматической аннотации центромер *CentromereArchitect*, который доступен по ссылке https://github.com/ablab/stringdecomposer/tree/ismb2021. По результатам было подготовленно две статьи. Первая часть была принята на международную конференцию "Intelligent Systems for Molecular Biology (ISMB) / European Conference on Computational Biology (ECCB) 2021 которая пройдет в июле и по результатам этой конференции будет опубликована статья статья в журнале Bioinformatics (импакт-фактор 5.610, Q1). Вторая часть работы будет подана в журнал после того, как первая статья будет опубликована и доступна в открытом доступе.

Список литературы

[1] Ahuja R. Magnati T. Orlin J. Network Flows: Theory, Algorithms, and Applications. — 1993.

[2] Alpha-CENTAURI: assessing novel centromeric repeat sequence variation with long read sequencing / Volkan Sevim, Ali Bashir, Chen-Shan Chin, Karen H Miga // Bioinformatics. — 2016. — Vol. 32, no. 13. — P. 1921–1924.

[3] Alpha-satellite DNA of primates: old and new families / Ivan Alexandrov, Alexei Kazakov, Irina Tumeneva et al. // Chromosoma. — 2001. — Vol. 110, no. 4. — P. 253–266.

[4] Annotation of suprachromosomal families reveals uncommon types of alpha satellite organization in pericentromeric regions of hg38 human genome assembly / VA Shepelev, LI Uralsky, AA Alexandrov et al. // Genomics data. — 2015. — Vol. 5. — P. 139–146.

[5] Black Elizabeth M, Giunta Simona. Repetitive fragile sites: centromere satellite DNA as a source of genome instability in human diseases // Genes. — 2018. — Vol. 9, no. 12. — P. 615.

[6] Bzikadze Andrey V, Pevzner Pavel A. centroFlye: assembling centromeres with long error-prone reads // BioRxiv. — 2019. — P. 772103.

[7] Classification and monomer-by-monomer annotation dataset of suprachromosomal family 1 alpha satellite higher-order repeats in hg38 human genome assembly / LI Uralsky, Valery Anatolyevich Shepelev, Alexandr Anatolyevich Alexandrov et al. // Data in brief. — 2019. — Vol. 24. — P. 103708.

[8] ColorHOR—Novel graphical algorithm for fast scan of alpha satellite higher-order repeats and HOR annotation for GenBank sequence of human genome / Vladimir Paar, Nenad Pavin, Marija Rosandić et al. // Bioinformatics. — 2005. — Vol. 21, no. 7. — P. 846–852.

[9] Compeau Phillip EC, Pevzner Pavel A, Tesler Glenn. How to apply de Bruijn graphs to genome assembly // Nature biotechnology. — 2011. — Vol. 29, no. 11. — P. 987–991.

[10] Davies David L, Bouldin Donald W. A cluster separation measure // IEEE transactions on pattern analysis and machine intelligence. — 1979. — no. 2. — P. 224–227.

[11] Dvorkina Tatiana, Bzikadze Andrey V, Pevzner Pavel A. The string decomposition problem and its applications to centromere analysis and assembly // Bioinformatics. — 2020. — Vol. 36, no. Supplement_1. — P. i93–i101.

[12] Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega / Fabian Sievers, Andreas Wilm, David Dineen et al. // Molecular systems biology. — 2011. — Vol. 7, no. 1. — P. 539.

[13] Henikoff Steven, Ahmad Kami, Malik Harmit S. The centromere paradox: stable inheritance with rapidly evolving DNA // Science. — 2001. — Vol. 293, no. 5532. — P. 1098–1102.

[14] Heterochromatin-encoded satellite RNAs induce breast cancer / Quan Zhu, Nien Hoong, Aaron Aslanian et al. // Molecular cell. — 2018. — Vol. 70, no. 5. — P. 842–853.

[15] Miga Karen H. Centromeric satellite DNAs: hidden sequence variation in the human population // Genes. — 2019. — Vol. 10, no. 5. — P. 352.

[16] Miga Karen H. Centromere studies in the era of 'telomere-to-telomere'genomics // Experimental cell research. — 2020. — P. 112127.

[17] Nagaoka So I, Hassold Terry J, Hunt Patricia A. Human aneuploidy: mechanisms and new insights into an age-old problem // Nature Reviews Genetics. — 2012. — Vol. 13, no. 7. — P. 493–504.

[18] Nurk S. Complete sequence of a human genome. — 2021.

[19] Organization and evolution of primate centromeric DNA from whole-genome shotgun sequence data / Can Alkan, Mario Ventura, Nicoletta Archidiacono et al. // PLoS Comput Biol. — 2007. — Vol. 3, no. 9. — P. e181.

[20] Satellite DNA evolution: old ideas, new approaches / Sarah Sander Lower, Michael P McGurk, Andrew G Clark, Daniel A Barbash // Current opinion in genetics & development. — 2018. — Vol. 49. — P. 70–78.

[21] Smurova Ksenia, De Wulf Peter. Centromere and pericentromere transcription: roles and regulation... in sickness and in health // Frontiers in genetics. — 2018. — Vol. 9. — P. 674.

[22] Suzuki Yuta, Myers Eugene W, Morishita Shinichi. Rapid and ongoing evolution of repetitive sequence structures in human centromeres // Science advances. — 2020. — Vol. 6, no. 50. — P. eabd9230.

[23] Waye John S, Willard Huntington F. Chromosome-specific alpha satellite DNA: nucleotide sequence analysis of the 2.0 kilobasepair repeat from the human X chromosome // Nucleic acids research. — 1985. — Vol. 13, no. 8. — P. 2731–2743.

[24] The evolutionary origin of man can be traced in the layers of defunct ancestral alpha satellites flanking the active centromeres of human chromosomes / Valery A Shepelev, Alexander A Alexandrov, Yuri B Yurov, Ivan A Alexandrov // PLoS Genet. — 2009. — Vol. 5, no. 9. — P. e1000641.

[25] The structure, function and evolution of a complete human chromosome 8 / Glennis A Logsdon, Mitchell R Vollger, PingHsun Hsieh et al. // Nature. — 2021. — P. 1–7.