Решение обобщенной задачи о связном подграфе максимального веса для сети Анализ обогащения

Александр Александрович Лобода[1]Артемов Максим Николаевич[2],
и Алексей Александрович Сергушичев[1] (B)

[1] Кафедра компьютерных технологий Университета ИТМО,
Санкт-Петербург 197101, Россия {
лобода, альсерг}@ rain.ifmo.ru
[2] Кафедра патологии и иммунологии Вашингтонского
университета в Сент-Луисе, Сент-Луис, Миссури, США
martyomov@pathology.wustl.edu

Абстрактный. Методы анализа обогащения сети позволяют идентифицировать активные модули без предвзятости *априори* определенные пути. Одна из математических формулировок такого анализа - сведение к задаче о связном подграфе максимального веса. В частности, при анализе метаболических сетей естественным образом возникает проблема обобщенного связного подграфа максимального веса (GMWCS), в котором оцениваются как вершины, так и ребра. Здесь мы представляем первый, насколько нам известно, практический точный решатель GMWCS. Мы протестировали его на реальных примерах и сравнили с аналогичными решателями. Во-первых, результаты показывают, что на экземплярах, взвешенных по узлам, решатель GMWCS имеет производительность, аналогичную лучшему решателю для этой проблемы. Во-вторых, решатель GMWCS работает быстрее по сравнению с ближайшим аналогом при запуске на экземплярах GMWCS с весами ребер.

Ключевые слова: Обогащение сети · Задача о связном подграфе с максимальным весом · Точный решатель · Смешанное целочисленное программирование

1 Введение

Методы обогащения набора генов широко используются для анализа нецелевых биологических данных, таких как транскриптомные, протеомные или метаболомные профили. Эти методы позволяют идентифицировать молекулярные пути в виде наборов генов, которые имеют неслучайное групповое поведение в данных. Определение таких избыточных путей дает представление о данных и позволяет лучше понять рассматриваемую систему.

Методы сетевого обогащения, в отличие от обогащения набора генов, не полагаются на предопределенные наборы генов и, таким образом, позволяют идентифицировать новые пути. Эти методы используют сеть взаимодействующих объектов, таких как гены, белки, метаболиты и т. Д., И пытаются идентифицировать наиболее регулируемую подсеть. Существуют разные математические формулировки задачи обогащения сети, но многие из них NP-трудны [1,6,9].

М. Фрит и CNS Pedersen (ред.): WABI 2016, LNBI 9838, стр. 210–221, 2016. DOI: 10.1007 /
978-3-319-43681-4 17

Dittrich et al. в [6] предложил формулировку проблемы связного подграфа максимального веса (MWCS). Первоначально авторы рассматривали взвешенный по узлам граф, в котором положительный вес соответствовал «интересным» узлам, а отрицательный вес - «неинтересным» узлам. Задача состояла в том, чтобы найти связный граф с максимальной суммой весов его узлов, соответствующий «активному модулю».

Здесь мы рассматриваем несколько иную форму MWCS: обобщенную MWCS (GMWCS), которая естественным образом возникает при изучении метаболических сетей [4,11]. В таких сетях узлы на графе представляют собой метаболиты, а ребра представляют их взаимопревращения посредством реакций. По сравнению с MWCS, GMWCS имеет также взвешенные края: узлы могут быть оценены с использованием метаболомных профилей, а края могут быть оценены с использованием профилей экспрессии генов или белков.

В последние годы огромная роль регуляции метаболизма становится все более и более признанной, особенно в контексте иммунной системы [10] и рак [5]. Это требует разработки эффективных вычислительных подходов для его изучения, таких как обогащение метаболической сети. Метод приводит к подсети связанных реакций, которые предположительно являются наиболее важными в рассматриваемом процессе. Используя такую подсеть, можно лучше понять соответствующую регуляцию метаболизма и, например, вывести ее критические точки [13].

В этой статье мы описываем точный решатель для задачи GMWCS, взвешенной по узлам и ребрам. Во-первых, в разд.2 дадим формальные определения. Затем в разд.3 мы описываем шаги предварительной обработки, адаптированные для краевой формулировки. В разд.4 мы покажем, как этот экземпляр можно разделить на три меньших экземпляра. Раздел5 посвящена формулировке задачи смешанного целочисленного программирования (MIP). В разд.6 мы показываем экспериментальные результаты запуска решателя на реальных экземплярах, которые появляются в веб-сервисе GAM, и показываем, что он быстрее и точнее, чем *Хайнц* [3] для экземпляров с взвешенными по краям экземпляров и по производительности аналогичен *Heinz2* [7] для экземпляров, взвешенных по узлам.

2 Формальные определения

Здесь мы рассматриваем проблему связного подграфа с максимальным весом (MWCS), для которой есть две несколько разные формулировки. В наиболее часто используемом определении MWCS взвешиваются только узлы [2,7]. В этой статье мы рассматриваем проблему, в которой ребра тоже взвешиваются [8]. Чтобы устранить двусмысленность, мы называем первую проблему Simple MWCS (SMWCS), а вторую - Generalized MWCS (GMWCS).

Задача задач MWCS - найти в заданном графе связный подграф с максимальной суммой весов. Поскольку подграф связан, мы можем рассматривать компоненты связности графа независимо. Таким образом, ниже мы предполагаем, что входной граф связан.

Во-первых, мы даем определение простой проблемы связного подграфа максимального веса.

Определение 1. *Учитывая связный неориентированный граф* $г$ знак равно V, E) *и весовая функция* ω_V: $V \to$ р, *простой связный подграф с максимальным весом (SMWCS) проблема заключается в нахождении связного подграфа* G знак равно $\tilde{V}, \tilde{E}$) *с максимальной полной массой*

$$\Omega(\text{Знак равно} \sum_{v \in \tilde{V}} \omega(v) \to \text{Максимум}$$

Во-вторых, мы определяем обобщенный вариант этой проблемы, в котором могут быть взвешены как узлы, так и ребра.

Определение 2. *Учитывая связный неориентированный граф* $г$ знак равно V, E) *и весовая функция* $\omega : (V \cup E) \to$ р, *обобщенный связный подграф максимального веса (GMWCS) проблема поиска связного подграфа* G знак равно $\tilde{V}, \tilde{E}$) *с максимальной полной массой*

$$\Omega(\text{Знак равно} \sum_{v \in \tilde{V}} \omega(v) + \sum_{e \in \tilde{E}} \omega(e) \to \text{Максимум}$$

Теперь мы определяем корневой вариант задачи с одной из вершин, вынужденных находиться в решении. Используется как вспомогательная подзадача GMWCS.

Определение 3. *Учитывая связный неориентированный граф* $г$ знак равно V, E), *весовая функция* $\omega : (V \cup E) \to$ р *и корневой узел* $p \in V$ *проблема корневого обобщенного связного подграфа максимального веса (R-GMWCS) - это проблема поиска связный подграф* G знак равно $\tilde{V}, \tilde{E}$) *такой, что* $p \in \tilde{V}$ *и*

$$\Omega(\text{Знак равно} \sum_{v \in \tilde{V}} \omega(v) + \sum_{e \in \tilde{E}} \omega(e) \to \text{Максимум}$$

Эль-Кебир и Клау в [7] показали, что проблема MWCS NP-сложна. Поскольку MWCS является частным случаем GMWCS, GMWCS также NP-сложен. Проблема R-GMWCS также является NP-сложной, потому что любой экземпляр проблемы GMWCS может быть решен путем решения экземпляра R-GMWCS для каждого узла в качестве корневого.

Наконец, ниже мы используем $п$ как сокращение количества узлов $/V/$ и $м$ по количеству ребер $/E/$ в графике $г$.

3 Предварительная обработка

Мы вводим два правила предварительной обработки, адаптированные из [7], которые упрощают задачу. Эти правила создают новый граф с меньшим числом вершин и ребер таким образом, чтобы решение GMWCS для исходного графа можно было легко восстановить из решения GMWCS для упрощенного графа.

Сначала мы объединяем группы близких вершин, которые либо ни одна, либо все не входят в оптимальное решение (рис. 1А). Позволять$е$ знак равноu, v) быть ребром с $\omega(e) \geq 0$ с одновременным $\omega(e) + \omega(v) \geq 0$ и $\omega(e) + \omega(ты) \geq 0$. В этом случае, если один из

вершины включаются в решение, тогда ребро и другая вершина также могут быть включены без уменьшения общего веса. Таким образом, мы можем сократить край e в новую вершину $ш$ с грузом $\omega(ш)$ знак равно $\omega(e) + \omega(ты) + \omega(v)$. После стягивания параллельные края между $ш$ и некоторая вершина $т$ может появиться. В этом случае мы объединяем все неотрицательные в одно ребро с весом, равным сумме их весов. После этого убираем все края между $ш$ и $т$ кроме одного с максимальным весом. Чтобы исчерпывающе применить правило в $O(м + кн)$ время, где k - количество сжатых ребер, мы можем использовать алгоритм 1.

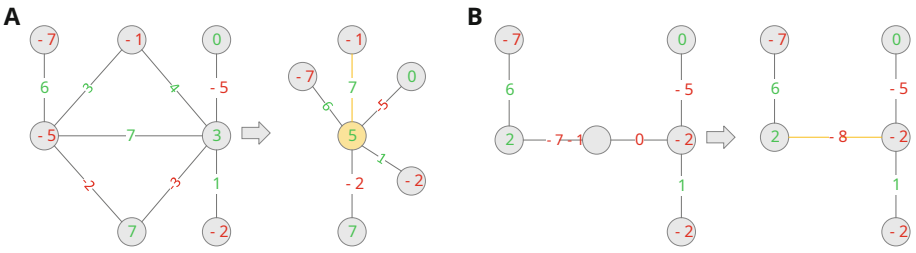

Рисунок 1. Применение первого правила, которое сокращает ребро (A), и второго правила, которое заменяет отрицательную цепочку одним ребром (B). Новые вершины и узлы окрашены в желтый цвет.

Алгоритм 1. Предварительная обработка усадки кромок

```
1: процедура ContractEdges (V, E) 2:
       для всех e ∈ E делать
3:         (u, v) ← e
4:         если ω(ты) + ω(e) <0 или ω(v) + ω(e) <0 тогда
5:             e ← нулевой
6:         пока e знак равно нулевой делать
7:             ш ← договор(e) e ←
8:             нулевой
9:             для всех z ∈ δш делать
10:                если ∃параллельные края e1, e2 между ш, г тогда
11:                    если ω(e1) ≥ 0 и ω(e2) ≥ 0 тогда
12:                        слить(e1, e2)
13:                    еще Удалить(arg min (ω(e')))
                                    e'∈ {e1, e2}
14:            для всех z ∈ δш делать
15:                e' ← (г, ш)
16:                если ω(ты) + ω(e') ≥ 0 и ω(v) + ω(e') ≥ 0 тогда
17:                    e ← e'
```

Во-вторых, аналогично предыдущему шагу объединяем неположительные цепочки (рис. 1Б). Позволять v быть вершиной с $град(v) = 2$ с соответствующими инцидентными ребрами e_1 знак равно u, v) и e_2 знак равно $v, и$). Если все три веса $\omega(v)$, $\omega(e_1)$ и $\omega(e_2)$ неположительны, то v, e_1 и e_2 можно заменить одной кромкой e знак равно $u, и$) с грузом $\omega(e$ знак равно $\omega(v) + \omega(e_1) + \omega(e_2)$. Объединение отрицательных цепочек реализовано в едином

пройти итеративно, пытаясь применить правило для всех узлов. Эта операция занимает $\Theta(n)$ время.

4 Разрезание вершин

В этом разделе мы обсудим, как экземпляр GMWCS можно разложить на три более мелкие проблемы. Декомпозиция основана на идее, что двусвязные компоненты можно рассматривать отдельно [7].

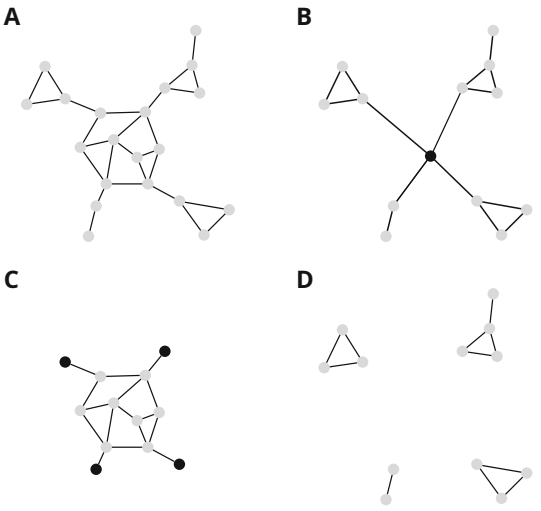

Рис. 2. Входной граф и экземпляры, порожденные декомпозицией

Вкратце, на входе у нас есть экземпляр GMWCS (рис. 2A). Сначала мы объединяем самый большой двусвязный компонент в одну вершину с нулевым весом и решаем экземпляр R-GMWCS для этого модифицированного графа и новой вершины как корня (рис.2Б). Затем мы заменяем каждую из составляющих, отходящих от наибольшего двусвязного компонента, на одну вершину с весом, равным весу соответствующего подграфа в решении R-GMWCS из предыдущего шага (рис.2C). Наконец, мы пытаемся найти подграф с большим весом, полностью лежащий в одной из компонент ветвления (рис.2D).

Формально пусть B быть двусвязным компонентом графа Γ с максимальным количеством вершин. Позволять C - множество разрезанных вершин графа Γ которые также содержатся в B. Позволять B_c быть компонентом, содержащим c в графике $\Gamma \setminus (ДО\ НАШЕЙ\ ЭРЫ)$.

Предложение 1. *Пусть подграф G графа Γ быть оптимальным решением GMWCS для графа Γ и G_c, $\forall c \in C$, оптимальные решения для экземпляров R-GMWCS для графов B_c с корнем c. В этом случае, если G содержит вершину $c \in C$, то мы можем построить оптимальное решение G' такая, что: (1) $G' \cap B = G \cap B$ и (2) $\Gamma' \cap B \tilde{=}_c \Gamma_c$.*

Доказательство. Позволять B_c знак равно $\Gamma \cap B_c$. Докажем, что его можно заменить на Γ_c без потери связи и оптимальности. Первый, B_c должен быть подключен. Пусть отключится. Тогда нет пути между c и некоторая вершина v. С Γ подключен, то есть простой путь vc в r. Однако по определению разрезаемой вершины путь vc не может содержать вершины из $G \setminus B_c$ и, таким образом, он полностью лежит в B_c, противоречие.

С B_c связан и содержит c тогда он не может иметь вес больше, чем Γ_c путем строительства Γ_c.

Теперь докажем, что замена сохраняет граф связным. Повторяя рассуждения из предыдущего шага, мы можем получить, что $\Gamma \cap B$ должен быть подключен. Так, Γ_c подключен, $\Gamma \cap B$ связаны, и оба эти графа содержат c. Таким образом, Γ' тоже связано.

Это предложение позволяет рассматривать только оптимальные решения, которые либо содержат вершину из B и в подграфах B_c идентичны соответствующему экземпляру R-GMWCS или полностью лежат в некоторых из подграфов B_c.

Во-первых, для каждого $c \in C$ мы хотим знать лучшее решение задачи для графа B_c содержащий вершину c. Это как раз экземпляр R-GMWCS. По практическим соображениям мы создадим один экземпляр на этом шаге вместо $|C|$ экземпляры. Позволять Γ^* знак равно $\forall c \in C\ B_c$. Затем мы объединяем все вершины из C содержалась в Γ^* в одну вершину p с участием $\omega(p) = 0$ и решить задачу R-GMWCS для такого графа. Позволять S быть решением этого случая. Чтобы получить решение для графика B_c заменяем обратно p к c в S, и удалим все вершины, не входящие в B_c.

Во-вторых, мы находим подграф с наилучшими оценками r что не полностью лежит в некоторых из B_c. Позволять Γ_c быть решением R-GMWCS для графа B_c с корнем c полученный на предыдущем шаге. Мы получаем новый экземпляр GMWCS, рассматривая компонент B и для всех $c \in C$ прикрепление вершины v с весом $\omega(v$ знак равно $\Omega(\Gamma_c)$. Мы решаем получившийся экземпляр, а затем восстанавливаем решение исходной проблемы.

Наконец, мы находим все потенциальные решения, которые полностью лежат в $\cup B_c$ для всех $c \in C$. Для этого мы создаем один экземпляр для графа r^* знак равно $\forall c \in C\ B_c$. Ясно, что если решение задачи для графа r полностью лежит в некоторых из B_c тогда мы найдем это на этом шаге.

Разложение графа на двусвязные компоненты требует $O(n + m)$ времени, генерация всех трех экземпляров также требует линейного времени, поэтому общая временная сложность на этом шаге равна $O(n + m)$.

5 Формулировка смешанного целочисленного программирования

Здесь мы описываем постановку задачи MIP. GMWCS может быть представлен как две части: целевая функция (вес подграфа), которая должна быть максимизирована, и ограничения, обеспечивающие связь подграфа.

Целевая функция является линейной и может быть легко помещена в задачу MIP. Однако получение эффективных ограничений связности линейных подграфов нетривиально. В этом разделе мы расскажем, как это можно сделать. Возникающая в результате проблема MIP решается с помощью IBM ILOG CPLEX.

Во-первых, мы рассматриваем нелинейную постановку задачи GMWCS, предложенную в [8]. Затем мы покажем, как устранить нелинейность и получить линейную систему. Наконец, мы вводим дополнительные нарушения симметрии и разрезы, которые не влияют на правильность формулировки, но улучшают производительность.

5.1 Представление подграфа

Мы используем одну двоичную переменную для каждой вершины или ребра, которые представляют присутствие в подграфе:

1. Двоичная переменная y_v принимает значение 1, если и только если $v \in V$ принадлежит подграфу.
2. Двоичная переменная $ш_e$ принимает значение 1, если и только если $e \in E$ принадлежит подграфу.

Чтобы эти переменные представляли действительный подграф (не обязательно связанный), нам нужно ввести набор ограничений:

$$ш_e \le y_v, \qquad \forall v \in V,\ e \in \delta_v. \tag{1}$$

Эти ограничения заявляют, что ребро может быть частью подграфа, только если обе его конечные точки являются частью подграфа.

5.2 Нелинейная формулировка

Нелинейная формулировка ограничений связности подграфа основана на идее, что любой связный граф может быть пройден из любой его вершины. Результат обхода можно представить как древообразование, где дуга (v, u) означает, что v был посещен раньше *ты.* Соответственно, мы можем гарантировать связность подграфа, если мы можем обеспечить древовидность, соответствующую обходу этого подграфа.

Для данного графа $г$ знак равно V, E), позволять S знак равно B, A) - ориентированный граф, где A получается из E заменяя каждую ненаправленную кромку e знак равно v, u) двумя направленными дугами (v, u) и (u, v).

Теперь мы собираемся ввести переменные, которые мы будем использовать в формулировке, и покажем нелинейную систему ограничений, обеспечивающих связность подграфа:

1. Двоичная переменная $Икс_a$ принимает значение 1, если и только если $a \in A$ относится к древовидным.
2. Двоичная переменная p_v принимает значение 1, если и только если $v \in V$ является корнем древесного происхождения.
3. Непрерывная переменная d_v принимает значение $п$ если путь в древовидной структуре от корня до вершины v содержит $п$ вершины. Если v не принадлежит решению, то значение может быть произвольным.

Затем мы вводим ограничения, обеспечивающие достоверность древовидной формы:

$$\sum_{v \in V} p_v \text{ знак равно } 1; \tag{2}$$

$$1 \le d_v \le п, \qquad \forall v \in V; \tag{3}$$

$$\sum_{(u,\, v) \in A} Икс_{уФ} + p_v \text{ знак равно } y_v, \qquad \forall v \in V; \tag{4}$$

$$Икс_{ву} + Икс_{уФ} \le Ше, \qquad \forall e \text{ знак равно } v,\, и) \in E; \tag{5}$$

$$d_v p_v \text{ знак равно } p_v, \qquad \forall v \in V; \tag{6}$$

$$d_{ты} Икс_{ву} \text{ знак равно } d_v + 1) Икс_{ву}, \qquad \forall (v,\, и) \in A. \tag{7}$$

Неравенство (2) утверждает, что в древовидной структуре только один корень; (3) - ограничение на расстояние между любой вершиной и корнем; (4) заявляет, что если вершина является частью подграфа, то либо она является корнем древовидной структуры, либо $град_в(v) = 1$; (5) говорит, что дуга древовидности может быть в растворе только в том случае, если соответствующее ребро также находится в ней. Последние два неравенства (6) и (7) контролировать правильные расстояния в древовидной структуре.

Haouari et al. показали в [8], что эта нелинейная система является правильной формулировкой GMWCS. То есть древовидность покрывает все вершины получившегося подграфа, и решение может вызвать это древообразование.

Однако неравенства (6) и (7) не являются линейными и должны быть заменены, чтобы формулировку можно было представить как задачу MIP.

5.3 Линеаризация

Нелинейные уравнения (6) и (7) можно заменить следующей системой линейных неравенств:

$$d_v + номер_v \le п, \qquad \forall v \in V; \tag{8}$$

$$п + d_{ты} - d_v \ge (п + 1) Икс_{ву}, \qquad \forall (v,\, и) \in A; \tag{9}$$

$$+ d_v - d_{ты} \ge (п - 1) Икс_{ву}, \qquad \forall (v,\, и) \in A. \tag{10}$$

Предложение 2. *Все возможные решения* (1) - (7) *также возможно* (1) - (5), (8) - (10) *и наоборот.*

Доказательство. Сначала докажем, что (8) эквивалентно (6) в смысле осуществимости решения. С p_v является двоичной переменной, мы можем рассмотреть два случая. Предположим, что p_v знак равно 1, тогда (6) примет вид d_v знак равно 1 пока (8) возьмет от $d_v \le 1$, а с (3) у нас есть d_v знак равно 1. Теперь предположим, что p_v знак равно 0, (6) будет выглядеть 0 = 0, это означает, что в этом случае нет дополнительных ограничений на переменные и (8) примет вид $d_v \le п$, но в системе уже есть такое неравенство. Таким образом (6) и (8) эквивалентны для обоих возможных значений p_v.

Во второй части доказательства мы воспользуемся тем же подходом. Здесь мы доказываем, что (7) можно представить в виде линейных неравенств (9) и (10).

1. Пусть $Икс_{ву}$ знак равно 1. Тогда после подстановки в (7) у нас есть $d_{ты}$ знак равно $d_v + 1$. Затем подставляем $Икс_{ву}$ в (9) и (10)

$$п + d_{ты} - d_v \geq п + 1 \; п$$
$$+ d_v - d_{ты} \geq п - 1$$

или, что то же самое,

$$d_{ты} \geq d_v + 1$$
$$d_v + 1 \geq d_{ты}$$

или $d_{ты}$ знак равно $d_v + 1$.

2. Пусть $Икс_{ву}$ знак равно 0. Исходное нелинейное уравнение примет вид 0 = 0. Как было сказано выше, это означает отсутствие дополнительных ограничений на переменные. Мы должны показать, что (9) и (10) тоже не добавляют таких ограничений. После подстановки эти неравенства принимают вид:

$$п + d_{ты} - d_v \geq 0 \; п$$
$$+ d_v - d_{ты} \geq 0$$

или $|d_v - d_{ты}| \leq п$. Очевидно, что переменные, которые имеют (3) автоматически выполняется такое неравенство. Таким образом, дополнительных ограничений не добавлено.

5.4 Нарушение симметрии

Обычной практикой является уменьшение количества возможных решений путем ограничения количества различных, но логически эквивалентных возможных решений. Такие решения называются симметричными. В нашей формулировке ограничения (1) - (5), (8) - (10) разрешить любое древообразование графа показать его связность. Итак, в этом разделе мы покажем, как уменьшить количество возможных древовидных образований и, таким образом, уменьшить пространство поиска.

Правило корневого порядка. Прежде всего, для некорневой задачи GMWCS мы заставляем корнем древовидности быть вершиной с максимальным весом среди присутствующих в подграфе. Соответствующее ограничение, добавленное в экземпляр MIP:

$$\sum_{v<ты} p_v \leq 1 - y_{ты}, \qquad \forall ты \in V, \tag{11}$$

где $v < ты$ если $\omega(v) < \omega(ты)$ или, если веса равны, мы используем некоторый фиксированный линейный порядок на вершинах.

Для R-GMWCS мы устанавливаем корень древовидной структуры таким же, как корень экземпляра.

Ограничение обхода. Более того, по связному графу можно перейти из одной вершины по-разному. Аналогично [12], мы показываем, как сделать невозможными такие решения, которые не могут быть достигнуты с помощью поиска в ширину (BFS).

Чтобы добиться такой формы древовидности, мы добавляем ограничения:

$$d_v - d_{ты} \leq п - (п - 1)ше, d_{ты} \qquad \forall e \text{ знак равно } v, и) \in Е; \qquad (12)$$

$$- d_v \leq п - (п - 1)ше, \qquad \forall e \text{ знак равно } v, и) \in Э. \qquad (13)$$

Эти ограничения утверждают, что если край *e* есть подграф, то расстояния до конечных точек отличаются на единицу.

Предложение 3. *Для любого связного подграфа* $г_s$ *графика* г *существует решение* (г, у, ш, х, г) *который кодирует подграф* $г_s$ *и возможно* (1) - (5), (8) - (10) *и* (11) - (13).

Доказательство. Во-первых, для любого подграфа $г_s$ мы можем выбрать любую из его вершин, в частности, с максимальным весом, и начать обход BFS, начиная с этой вершины. Как было показано выше для любого связного подграфа$г_s$ и любой его древовидности есть соответствующая кодировка (г, у, ш, х, г), удовлетворяющие ограничениям (1) - (5) и (8) - (10). Путем выбора вершины с максимальным весом в качестве ограничения корня древовидности (11) имеет место. Ограничения (12) - (13) также выполняются, поскольку они прямо следуют из порядка BFS.

6 Экспериментальные результаты

В качестве тестового набора данных мы использовали 101 экземпляр, созданный Shiny GAM, веб-сервисом для интегрированного анализа транскрипции и метаболической сети [11] на основе данных, отправленных пользователями на этапе тестирования. В наборе данных есть 38 экземпляров SMWCS со взвешенными узлами и 63 экземпляра GMWCS. Архив с экземплярами доступен по адресуhttp://genome.ifmo.ru/files/papers files / WABI2016 / gmwcs / instance.tar.gz. Вкратце, взвешенные по узлам экземпляры содержат около 2200 узлов и 2500 ребер и соответствуют сети с узлами как для метаболитов, так и для реакций, которые связаны, если метаболит является субстратом или продуктом реакции. Экземпляры, взвешенные по краям, содержат около 700 узлов и 900 ребер. Метаболиты и реакции оцениваются пропорционально логарифму соответствующих p-значений дифференциального выражения.

Для сравнения мы выбрали два других решателя: *Хайнц* версия 1.68 [6] и *Heinz2* версия 2.1 [7]. Первый,*Хайнц*, изначально был разработан для SMWCS со взвешенными узлами, но позже был скорректирован с учетом весов ребер, однако рассматриваются только ациклические решения. Второй,*Heinz2*, не принимает веса кромок, но работает быстрее, чем *Хайнц* на экземплярах, взвешенных по узлам.

Мы запускали каждый решатель на каждом из экземпляров по 10 раз с ограничением по времени 1000 с. *Heinz2* и наш решатель GMWCS были запущены с использованием 4 потоков. Процессором был AMD Opteron 6380 2,5 ГГц. Таблица с таблицей результатов доступна по адресуhttp://genome.ifmo.ru/files/papers files / WABI2016 / gmwcs / results.final.tsv.

6.1 Результаты для простого MWCS

Эксперименты показали, что на экземплярах, взвешенных по узлам, решатель GMWCS имеет производительность, аналогичную *Heinz2* (Инжир. 3A). Для 24 экземпляров (63%) GMWCS работает медленнее, чем *Heinz2*. Однако 32 случая (84%) были решены GMWCS в течение 30 секунд, по сравнению с 27 (71%) случаями. *Heinz2*. Более того, 4 случая не были раскрыты *Heinz2* за допустимое время 1000 с по сравнению с одним экземпляром для GMWCS.

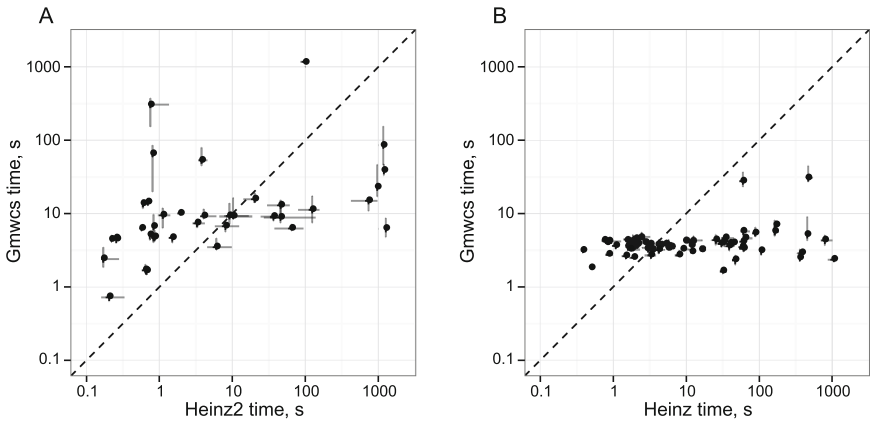

Рис 3. Сравнение GMWCS с *Heinz2* и *Хайнц* решатели на экземплярах, взвешенных по узлам (A) и по узлам и ребрам (B). Точки представляют собой среднее время 10 прогонов на одном экземпляре. Горизонтальные и вертикальные серые линии представляют второе минимальное и второе максимальное время. Для удобства к медианным значениям более 950 с был добавлен небольшой случайный шум.

6.2 Результаты для обобщенного MWCS

Для экземпляров GMWCS, взвешенных по краям, решатель GMWCS смог найти оптимальные решения в течение 10 секунд во всех экземплярах, кроме двух, в то время как это заняло *Хайнц* более 10 с на решение 30 случаев (48%) (рис. 3Б). Более того, только 35 экземпляров (56%) имели ациклическое решение, соответственно 28 экземпляров не были решены до GMWCS-оптимальности с помощью *Хайнц*.

7 Заключение

Для анализа биологических данных активно развиваются подходы сетевого анализа. С математической точки зрения это обычно NP-сложные задачи. Здесь мы описали точное практическое решение для конкретной постановки обобщенной проблемы связного подграфа максимального веса, которая естественным образом возникает в метаболических сетях. Мы протестировали метод на реальных данных и показали, что разработанный решатель аналогичен по производительности.

к существующему решателю *Heinz2* на простых экземплярах MWCS и работает лучше и точнее по сравнению с *Хайнц* на экземплярах, взвешенных по ребрам. Реализация находится в свободном доступе по адресуhttps://github.com/ctlab/gmwcs-solver.

Финансирование. Работа поддержана Правительством Российской Федерации [грант 074-U01 для AAS, AAL].

использованная литература

1. Алькарас, Н., Полинг, Дж., Батра, Р., Барбоса, Э., Юнге, А., Кристенсен, AGL, Азеведо, В., Дицель, Х.Дж., Баумбах, Дж.: KeyPathwayMiner 4.0: путь к конкретным условиям анализ путем объединения нескольких исследований и сетей omics с cytoscape. BMC Syst. Биол.**8**(1), 99 (2014)
2. Альварес-Миранда, Э., Любич, И., Муцель, П .: Проблема связного подграфа максимального веса. В: Jünger, M., Reinelt, G. (eds.) Festschrift for Martin Grötschel, стр. 245–270. Спрингер, Гейдельберг (2013)
3. Байссер, Д., Брюнкхорст, С., Дандекар, Т., Клау, Г.В., Диттрих, М.Т., Мюллер, Т .: Надежность и точность функциональных модулей в комплексном сетевом анализе. Биоинформатика**28 год**(14), 1887–1894 (2012). (Оксфорд, Англия)
4. Байссер, Д. и др.: Интегрированные модули путей, использующие профили метаболизма с течением времени и данные EST из Milnesium tardigradum. BMC Syst. Биол.**6**, 72 (2012)
5. Кэрнс, Р.А., Харрис, И.С., Мак, Т.В.: Регулирование метаболизма раковых клеток. Nat. Преподобный Рак**11**2. С. 85–95 (2011).
6. Диттрих, М.Т., Клау, Г.В., Розенвальд, А., Дандекар, Т., Мюллер, Т .: Идентификация функциональных модулей в сетях взаимодействия белок-белок: интегрированный точный подход. Биоинформатика**24**(13), i223 – i231 (2008). (Оксфорд, Англия)
7. Эль-Кебир, М., Клау, GW: Решение проблемы связного подграфа максимального веса до оптимальности (2014). arXiv: 1409.5308
8. Haouari, M., Maculan, N., Mrad, M .: Расширенные компактные модели для проблемы связного подграфа и для проблемы кратчайшего пути в орграфах с отрицательными циклами. Comput. Опер. Res.**40**(10), 2485–2492 (2013).
9. Ideker, T., Ozier, O., Schwikowski, B., Siegel, AF: открытие регуляторных и сигнальных цепей в сетях молекулярного взаимодействия. Биоинформатика**18**(Приложение 1), S233 – S240 (2002). (Оксфорд, Англия)
10. Матис, Д., Шелсон, С.Е .: Иммунометаболизм: новые рубежи. Nat. Rev. Immunol. **11**(2), 81 (2011)
11. Сергушичев, А., Лобода, А., Джа, А., Винсент, Э., Дриггерс, Э., Джонс, Р., Пирс, Э., Артемов, М .: GAM: веб-сервис для комплексного анализа транскрипционной и метаболической сети. Nucleic Acids Res. (2016).http://nar.oxfordjournals.org/citmgr? type = bibtex & gca = nar% 3Bgkw266v1
12. Ульянцев В., Закирзянов И., Шалыто А .: Предикаты нарушения симметрии на основе BFS для идентификации DFA. В: Дедиу, А.-Х., Форменти, Э., Мартин-Виде, К., Труте, Б. (ред.) LATA 2015. LNCS, vol. 8977, стр. 611–622. Спрингер, Гейдельберг (2015)
13. Винсент, Э.Е. и др.: Митохондриальная фосфоенолпируваткарбоксикиназа регулирует метаболическую адаптацию и обеспечивает независимый от глюкозы рост опухоли. Мол. Клетка **60**2. С. 195–207 (2015).