

Чисел Кармайкла бесконечно много

Числа Кармайкла

Для любого числа a и простого n $a^n \bmod n = a$. Но это может быть верно и для составных n . Такие n и называются числами Кармайкла.

Критерий Корселя

Число является числом Кармайкла тогда и только тогда, когда :

- 1) Оно свободно от квадратов.
- 2) Для любого p - простого делителя n $p - 1$ делит $n - 1$.

Результаты аналитической теории чисел

Теорема 1

Пусть $\pi(x)$ - количество простых, не превосходящих x , $\pi(x, y)$ - количество простых $p \leq x$, таких что все простые делители числа $p - 1$ не превосходят y . Тогда существует маленькая константа E , такая что $\pi(x, x^{1-E}) \geq C_E \pi(x)$. Гипотеза состоит в том, что любое число до 1 подходит на роль этой константы.

Теорема 2

Пусть $\pi(x, d, a)$ - количество простых чисел до x , сравнимых с a по модулю d . Тогда существует очень маленькое число B с таким свойством: Пусть x достаточно большое, $1 \leq d \leq \min(x^B, \frac{y}{x^{1-B}})$. Тогда $\pi(x, d, a) \geq \frac{\pi(y)}{2\phi(d)}$, если d не делится на множество чисел фиксированного размера, не меньших $\log x$. Это значит, что во многих арифметических прогрессиях с не очень большим знаменателем простых чисел достаточно много. Гипотеза опять же состоит в том, что любое число до 1 подходит на роль этой константы.

ТЕОРЕМА

Пусть $C(x)$ - количество чисел Кармайкла, не превосходящих x . Тогда для любого $\epsilon > 0$ $C(x) > x^{BE-\epsilon}$ при достаточно больших x . Гипотезы дают $C(x) > x^{1-\epsilon}$ для любого ϵ , подстановка наилучших известных доказанных значений для B и E дает чуть-чуть больше чем $\frac{2}{7}$.

Лемма 1

Пусть G - конечная абелева группа, $n(G)$ - минимальное число с таким свойством - из любых $n(G)$ чисел можно выбрать несколько (не 0) с произведением 1. Тривиальная оценка - $n(G) < |G| + 1$, но для групп, далеких от циклической, можно и точнее. А именно, пусть m - максимальный порядок элемента группы, тогда $n(G) < m(1 + \log \frac{|G|}{m})$

Лемма 2

Пусть G - абелева группа, $r > t > n = n(G)$. Тогда из любых r элементов группы можно выбрать от $t - n$ до t с произведением 1, причем это можно сделать не менее чем $\frac{C_r^t}{C_r^n}$ способами.

Для начала мы хотим построить такое число L , которое имеет много делителей вида $p - 1$ для простого p . Идейно это потому, что: мы найдем число n с просто большим числом делителей, а потом для каждого делителя d рассмотрим числа $dk + 1$ для небольших k . Аналитические леммы обеспечат нам, что для каждого делителя среди них много простых. Тогда существует k , для которого среди чисел $dk + 1$ много простых, а dk - делитель числа nk .

Далее мы хотим выбрать из этих делителей много наборов p_1, p_2, \dots, p_k , такой что $M = p_1 p_2 \cdots p_k \equiv 1 \pmod L$, пользуясь комбинаторными леммами (они хорошо работают как раз с числами с большим числом маленьких делителей). Тогда $M - 1$ делится на L , а L на $p - 1$, и число M - число Кармайкла.

Я буду рассказывать о r -дифференциальных частично упорядоченных множествах. Что такое частично упорядоченное множество все должны знать.

Определение 1. Частично упорядоченное множество P называется локально конечным, если $\forall a, b \in P$ таких $x \in P$, что $a < x < b$ конечное число.

Определение 2. Частично упорядоченное множество P называется градуированным, если оно снабжено функцией ранга $\rho : P \rightarrow \mathbb{N}_0$:

1) Если $x < y$, то $\rho(x) < \rho(y)$.

2) Если $x < y$ и $\nexists z : x < z < y$, то $\rho(y) = \rho(x) + 1$. В таком случае говорят, что y покрывает x .

Главное определение:

Определение 3. Пусть $r \in \mathbb{N}$ и $P = \cup_{n \geq 0} P_n$ локально конечное градуированное частично упорядоченное множество с наименьшим элементом нулевого ранга. Тогда оно называется r -дифференциальным, если:

1) $\forall x \neq y \in P$ покрывают k общих элементов \Leftrightarrow они покрываются k общими элементами.

2) $x \in P$ покрывает m элементов \Leftrightarrow он покрывается $m+r$ элементами.

Утверждения:

Утверждение 1. В первой части крайнего определения k может принимать только значения 0 и 1.

Определение 4. Размерность гиперребра F гиперграфа H - $\dim F = |F| - 1$.

Утверждение 2. $\forall r$ существует биекция между $P_{[1,2]} = P_1 \cup P_2$ с гиперграфами с множеством вершин $V = \{1, 2, \dots, r\}$, такими что размерность всех рёбер положительна, и любое подмножество V из двух элементов содержится ровно в одном гиперребре.

Утверждение 3. Пусть P - r -дифференциальное ЧУМ с ранг-функцией $\rho : \rho_0 = 1, \rho_1 = r, \dots (\rho_i = |P_i|)$, а T_1 - сумма размерностей гиперрёбер гиперграфа, ассоциированного с $P_{[1,2]}$. Тогда:

$$\rho_2 = r(r+1) - T_1.$$

Утверждение 4. Для r -дифференциального ЧУМ максимальное возможное значение ρ_2 равно $r^2 + 1$, следующее равно $r^2 - r + 3$, а минимальное равно $\frac{r^2+3r}{2}$.

Утверждение 5. $r \geq 6 \Leftrightarrow \exists P, Q : P$ и Q r -дифференциальные ЧУМ, а P и Q имеют одинаковые ранг-функции, но $\forall a \in \mathbb{N} P_{[a,a+1]} = P_a \cup P_{a+1}$ не изоморфно $Q_{[a,a+1]} = Q_a \cup Q_{a+1}$.

Определение 5. Говорят, что множество последовательностей натуральных чисел удовлетворяет интервальному свойству, если:

$\forall h = (h_0, \dots, h_{i-1}, h_i, h_{i+1}, \dots), h' = ((h_0, \dots, h_{i-1}, h_i + \alpha, h_{i+1}, \dots))$, где $\alpha > 0$ - последовательностей из этого множества любая последовательность вида $(h_0, \dots, h_{i-1}, h_i + \beta, h_{i+1}, \dots)$, где $0 < \beta < \alpha$ принадлежит этому множеству.

Теорема 1. Множество всех ранг функций r -дифференциальных ЧУМов не удовлетворяет интервальному свойству.

Определение 6. Пусть $P = \cup P_{n \geq 0}$ - r -дифференциальное ЧУМ.

$\kappa(n \rightarrow n+1 \rightarrow n \rightarrow n+1 \rightarrow n) :=$ количество путей вида $x_1 < x_2 > x_3 < x_4 > x_5 = x_1$, где $x_1, x_3 \in P_n$, $x_2, x_4 \in P_{n+1}$.

$\alpha(n \rightarrow n+1 \rightarrow n) :=$ количество путей вида $x_1 < x_2 > x_3$, где $x_1, x_3 \in P_n$, $x_2 \in P_{n+1}$.

$\alpha(n \rightarrow n+1) :=$ количество путей вида $x_1 < x_2$, где $x_1 \in P_n$, $x_2 \in P_{n+1}$.

Для $x \in P$ $c(x) :=$ количество элементов, покрывающих x .

Лемма 1. $\forall n \geq 0$ и P - r -дифференциального ЧУМа

$$\sum_{x \in P_n} c(x)^2 = \kappa(n \rightarrow n+1 \rightarrow n \rightarrow n+1 \rightarrow n) - \alpha(n \rightarrow n+1 \rightarrow n) + \alpha(n \rightarrow n+1).$$

Лемма 2. $\forall n \geq 0$ и P - r -дифференциального ЧУМа с ранг-функцией p

$$\sum_{x \in P_n} c(x)^2 = \sum_{j=0}^n (r^2(n-j+1) + \epsilon r) p_j,$$

где $\epsilon = (n-j) \bmod 2$.

Теорема 2. $\forall P$ - r -дифференциального ЧУМа с ранг-функцией $p \exists a :$

$$n^a e^{2\sqrt{rn}} = O(p_n).$$

Всё доказывается либо ручками, либо ссылками на другие статьи. Я в этом не виноват, это статья такая.

19.04

На этой паре мы занимались тем, что ввели определения, обсудили связь между ними и попытались мотивировать изучение эвристик из класса $1\text{DSPACE}'$.

Определение 1. *Машина Тьюринга называется онлайн машиной Тьюринга, если она может двигать (но не обязана) головку на входной ленте только в одном направлении.*

Определение 2. *Класс всех языков, распознаваемых онлайн машиной Тьюринга, использующей не более $f(n)$ памяти, будем обозначать $1\text{DSPACE}(f)$.*

Определение 3. *Класс языков, распознаваемых онлайн машиной Тьюринга, которая может узнать длину входа и использует не более $f(n)$ памяти, будем обозначать $1\text{DSPACE}'(f)$.*

Лемма 1. *Существует язык $L \in 1\text{DSPACE}'(\log n)$ такой, что $L \notin 1\text{DSPACE}(o(n))$.*

Здесь в качестве L подходит множество всех таких двоичных слов s , что двоичная запись $|s|$ является некоторым префиксом s .

Лемма 2. *Для любой функции f такой, что $f(n) < \frac{n}{2}$ при всех достаточно больших n , и f может быть вычислено с использованием $O(\log f)$ памяти, существует язык L такой, что $L \in \text{DSPACE}(\log f) \cap 1\text{DSPACE}'(f)$ и $L \notin 1\text{DSPACE}'(o(f))$.*

Тут подходит множество всех f -периодичных строк.

В лемме 2 у нас возникла трудность с вычислением f и на семинаре я предложил просто взять $f = n/4$, но после мы выяснили, что в качестве f подходит любая достаточно разумная функция, просто нужно внимательно следить за определениями. Подробнее в разделе с замечаниями.

Основным результатом была такая теорема.

Теорема 1. *Для любой функции $f = \Omega(\log \log n)$ и языка L , распознаваемого оффлайн машиной Тьюринга M с использованием $f(n)$ памяти и рабочим алфавитом Γ , $L \in 1\text{DSPACE}'(f(n) \cdot |\Gamma|^{f(n)})$, если $f(n)$ может быть вычислено с использованием $O(f(n) \cdot |\Gamma|^{f(n)})$ памяти.*

Доказательство этой теоремы очень похоже на сведение двусторонних конечных автоматов к односторонним (обычным), с одной небольшой тонкостью, связанной с зацикливанием. Дело в том, что машина

Тьюринга, даже если она останавливается на любом входе, может за-
 циклиться, если ее запустить из неправильной конфигурации. В авто-
 матах такое тоже бывает, но там об этом можно не думать, поскольку
 все функции переходов можно вычислить заранее (они же конечные!). Я
 эту проблему обходил при помощи техники baby-step giant-step (то есть
 запуска двух симуляций с разными скоростями), но на самом деле ее
 можно решать как угодно, например, просто добавлением счетчика (но
 тогда надо внимательно следить за памятью).

Algorithm 1 DPLL

```

1: procedure DPLLA,B( $\varphi$ )
2:   if  $\varphi$  is empty then
3:     return satisfiable
4:   if  $\varphi$  contain empty clause then
5:     return unsatisfiable
6:    $x \leftarrow A(\varphi)$ 
7:    $b \leftarrow B(\varphi, x)$ 
8:   if DPLLA,B( $\varphi[x = b]$ ) = satisfiable then
9:     return satisfiable
10:  return DPLLA,B( $\varphi[x = \neg b]$ )

1: procedure DPLLH( $\varphi$ )
2:   if  $\varphi$  is empty then
3:     return satisfiable
4:   if  $\varphi$  contain empty clause then
5:     return unsatisfiable
6:    $(x, b) \leftarrow H(\varphi)$ 
7:   if DPLLH( $\varphi[x = b]$ ) = satisfiable then
8:     return satisfiable
9:   return DPLLH( $\varphi[x = \neg b]$ )
    
```

Еще мы определили два вида DPLL (классический с двумя эвристи-
 ками, а нужный для наших целей — с одной) и поняли, что они друг от
 друга в терминах сложности по памяти почти ничем не отличаются.

26.04

Мы всю пару доказывали экспоненциальную нижнюю оценку на DPLL_H
 с $H \in 1\text{DSPACE}'(o(\frac{n}{\log n}))$. Подробности напишу позже.

Общие замечания

Здесь чуть позже появятся замечания, которые могут быть интересны тем, кто был на семинаре.

В докладе будут использоваться более-менее стандартные графские обозначения, например, $H-v-X$ для обозначения графа H , из которого удалили вершину $v \in V(H)$ и подмножество вершин $X \subset V(H)$. Расстояние между вершинами x и y в графе H будем обозначать как $d_H(x, y)$. Очевидно, $d_H(x, y) = d_H(y, x)$ для неориентированного графа H .

Задача о самой важной вершине – дан связный взвешенный (веса всех рёбер положительные) неориентированный граф G и две вершины в нём: s и t . Хотим удалить вершину G , отличную от s и t так, чтобы максимизировать расстояние между s и t в оставшемся графе (если пути между s и t в оставшемся графе нет, то положим расстояние равным $+\infty$).

Наивный алгоритм нахождения такой вершины работает за $\Theta(nm \log n)$ в графе с n вершинами и m рёбрами: перебираем удаляемую вершину v ($n - 2$ варианта) и ищем кратчайший путь от s до t в графе $G-v$ (алгоритм Дейкстры, требует $\Theta(m \log n)$ времени в самой простой реализации).

Наш алгоритм будет иметь время работы $O(m \log n)$ или даже $O(m + n \log n)$, если использовать фиббоначиевые кучи. Более того, он для каждой вершины графа v графа посчитает длину кратчайшего пути между s и t в графе $G - v$, что даёт намного больше информации, чем просто номер самой важной вершины.

Алгоритм основан на нескольких несложных наблюдениях, но, как говорил, Сергей Витальевич, “Две пропущенные подряд тривиальности образуют непреодолимое препятствие”.

Def. Деревом кратчайших путей связного взвешенного графа H с центром в вершине $r \in V(H)$ называется такое любое такое корневое дерево T на множестве вершин $V(H)$, что

- r — корень T .
- Все рёбра T являются рёбрами исходного графа H .
- Для каждой вершины $v \in V(G)$ верно, что $d_G(r, v) = d_T(r, v)$.

Lemma. Пусть H — связный взвешенный граф, а $v \in V(H)$ — его вершина. Тогда у H есть дерево кратчайших путей с корнем в v .

Lemma. Пусть T — какое-то дерево кратчайших путей для H . Тогда $d_T(u, v) = d_H(u, v)$, если u является предком v в дереве T .

Рассмотрим у графа G какое-нибудь дерево кратчайших путей T с корнем в s , а также нисходящий путь $P = (v_0, v_1, \dots, v_k)$ от s к t в этом дереве, где $v_0 = s$ и $v_k = t$. По определению дерева кратчайших путей путь P кратчайший в G . Поэтому $d_{G-u}(s, t) = d_G(s, t)$, если $u \notin P$.

Поняли, что стоит удалять только вершины из P . Пусть $i \in [0, k-1]$. Введём такие обозначения:

- U_i — пустое множество вершин при $i = 0$, иначе компонента связности графа $T - v_i$, содержащая вершину v_{i-1} , то есть вершины “сверху” от v_i .
- D_i — поддереву вершины v_{i+1} в дереве T , то есть вершины “снизу” от v_i (“низ” — направление к t).
- O_i — все вершины поддереву вершины v_i в дереве T , за исключением самой вершины v_i и всех вершин из множества D_i . В некотором смысле, вершины находящиеся “сбоку”.

Lemma. $d_G(s, u) = d_{G-v_i}(s, u)$ для каждой вершины $u \in U_i$

Несложно видеть, что любой путь из s в t , не проходящий через v_i , где $i \in [1, k-1]$ выглядит так: сперва он как-то ходит по $U_i \cup O_i$, а потом впервые заходит в D_i и уже из какой-то вершины D_i приходит в t . Оказывается, верна такая лемма:

Lemma. $d_G(u, t) = d_{G-v_i}(u, t)$ для каждой вершины $u \in D_i$. То есть в G существует кратчайший путь между u и t , не проходящий через v_i .

То есть каждый интересующий нас путь из s в t в графе $G - v_i$ выглядит так: сперва он как-то ходит по $U_i \cup O_i$, приходит в некоторую вершину $x \in U_i \cup O_i$, потом проходит по какому-то ребру $(x, y) \in E(G)$ в вершину $y \in D_i$, а потом проходит из y в t по пути длины $d_G(y, t)$.

Таким образом, $d_{G-v_i}(s, t) = \min d_{G-v_i-D_i}(s, x) + w(x, y) + d_G(y, t)$, где минимум берётся по всем таким рёбрам $(x, y) \in E(G)$, что $x \in U_i \cup O_i, y \in D_i$.

Величину $d_{G-v_i-D_i}(s, x)$ можно посчитать так: это $d_G(s, x)$, если $x \in U_i$, а для $x \in O_i$ её можно посчитать с помощью алгоритма Дейкстры: можно понять, что кратчайшему пути из s в $x \in O_i$ в графе $G - v_i - D_i$ нет смысла снова заходить в U_i после того, как он уже зашёл в O_i . Поэтому нужно просто сперва положить $d_{G-v_i-D_i}(s, x) = \min d_{G-v_i-D_i}(s, z) + w(z, x) = \min d_G(s, z) + w(z, x)$, где минимум берётся по всем таким рёбрам $(z, x) \in E(G)$, что $z \in U_i$. После этого нужно применить обычный алгоритм Дейкстры на вершинах O_i и рёбрах, оба конца которых принадлежат O_i . Эта стадия алгоритма работает за $O(m \log n)$ суммарно и позволяет насчитать $O(n)$ информации, позволяющей за $O(1)$ отвечать на запрос “чему равно $d_{G-v_i-D_i}(s, x)$ ”.

Наконец, нужно как-то посчитать эти минимумы для каждого i , вообще говоря каждый минимум может браться по $O(m)$ рёбрам, поэтому наивным образом этот минимум считать нельзя. Однако можно заметить, что при переходе от v_i к v_{i+1} множество рёбер, по которым берётся минимум, изменилось не сильно:

- Удалились рёбра с одним концом в множестве $U_i \cup O_i$, а другим в множестве $D_i \setminus D_{i+1} = O_{i+1} \cup \{v_{i+1}\}$.
- Добавились рёбра с одним из концов в множестве $\{v_i\} \cup O_{i+1} = (U_{i+1} \cup O_{i+1}) \setminus (U_i \cup O_i)$ и другим в множестве D_{i+1} .

- Наконец, рёбра с одним концом в множестве O_i , а другим в множестве D_{i+1} стали вносить другой вклад в ответ: раньше они вносили $d_{G-v_i-D_i}(s, x) + w(x, y) + d_G(y, t)$, а теперь $d_{G-v_{i+1}-D_{i+1}}(s, x) + w(x, y) + d_G(y, t)$. Разница в том, что $d_{G-v_i-D_i}(s, x)$ для $x \in O_i$ мы считали отдельно с помощью Дейкстры, а $d_{G-v_{i+1}-D_{i+1}}(s, x) = d_G(s, x)$.

Мы хотим иметь множество рёбер, в котором мы можем удалять часть имеющихся рёбер, менять части имеющихся ребёр “вклад” и добавлять немного новых рёбер и, наконец, спрашивать минимальный “вклад” ребра из множества (чтобы узнать $\min d_{G-v_i-D_i}(s, x) + w(x, y) + d_G(y, t)$, то есть $d_{G-v_i}(s, t)$). Любая куча способна поддерживать такие операции. Если аккуратно посчитать, то всего операций, когда i пробегает значения от 1 до $k - 1$, будет $O(m)$, поэтому эта часть алгоритма работает за $O(m \log n)$.

Проблема дискретной геодезической

Alex Morakhovski

April 2019

1 Introduction

Нам дан многогранник, заданный набором граней, ребер и вершин. Мы считаем грани замкнутыми многоугольниками (они включают в себя их границы) и ребра должны быть отрезками отрезков (они включают в себя их конечные точки, которые являются вершинами). Нам также даны две специальные точки s и t , начало и конец пути. Без ограничения общности мы предполагаем, что все грани - это треугольники и что s и t вершины многогранника. Нас просят найти кратчайший путь от источника к месту назначения, которое полностью лежит на поверхности.

Утверждение 1 *Существует геодезическая от s до любой другой точки x . Кроме того, среди геодезических от s до x существует по крайней мере один путь минимальной длины.*

Утверждение 2 *Если p является геодезической, которая соединяет последовательность ребер, то плоское разворачивание вдоль последовательности ребер представляет собой отрезок.*

Утверждение 3 *Общая форма геодезической - это путь, который проходит через чередующиеся последовательности вершин и (возможно пустые) реберные последовательности, так что развернутое изображение пути вдоль любой последовательности ребер представляет собой отрезок и угол пути, проходящего через вершину, больше или равен π .*

Утверждение 4 *Оптимальный путь $p(x)$ к точке x проходит через внутренность не более одной грани содержащего x .*

2 Алгоритм

Алгоритм работает примерно как "непрерывный" Алгоритм Дейкстры. "Сигнал" распространяется от источника к остальной поверхности. Как только точка x поверхности получает сигнал в первый раз, она распространяется дальше; точка x считается постоянно помеченной временем $d(x)$, в которое она получила сигнал, что, разумеется, является минимальным расстоянием

от источника до x . К счастью, мы должны сделать эту маркировку и перераспространение только для конечного числа (фактически, как мы покажем, $O(n^2)$) точек (называемых точками события) поверхности. (На самом деле мы делаем "направленную" форму непрерывной Дейкстры, поскольку мы помечаем точки на ребре с длинами путей, падающих на ребро с любой из двух сторон ребра.) Алгоритм использует несколько простых структур данных. Мы ведем список, $PLIST$, кандидатов интервалов оптимальности. Интервал-кандидат (или короткий интервал) - это подсегмент ребра, который является суперсегментом некоторого (возможно, пустого) интервала оптимальности и имеет точно такую же структуру, что и этот интервал (т. е. Он имеет тот же тип информации, связанный с ним в его структуре данных). Мы называем их интервалами кандидатами, потому что при завершении нашего алгоритма все оставшиеся будут нашими интервалами оптимальности геодезической.

3 Алгоритм еще раз

(0)- Инициализация. Помечаем s нулем. Инициализируем $PLIST$, чтобы быть пустым. Для каждого ребра, противоположного s , создайте интервал кандидата, длина которого равна всему ребру, а корень s . Вычисляем точку самую близкую к s на каждом из противоположных ребер, помещаем каждую такую точку и конечные точки ребер в очередь событий, причем каждая точка помечает свое расстояние от s . Вставляем полученные интервалы в список $PLIST$ и в списки интервалов соответствующих пар ребер-граней.

(1)-Main Loop. Пока в очереди событий есть запись, удаляем самое маленькое значение и помечаем его. Если он помечен как точка границы некоторого потенциального интервала, тогда делаем Распространение. Интуитивн, распространение интервала означает, что "волна" сигналов от корня проходит через интервал к другим двум краям грани.

Если какие то точки покрываются двумя интервалами, то удаляем ту с большим значением.

Алгоритм работает, и работает с временем $O(n \log n)$ и требует $O(n^2)$ памяти, где n - количество ребер поверхности. После того, как мы запустим наш алгоритм, расстояние от источника до любого другого пункта назначения может быть определено с использованием стандартных методов за время $O(\log n)$

Теорема об изоморфизме деревьев

Определение 1. A и B – два дерева с множествами вершин a_1, a_2, \dots, a_n и b_1, b_2, \dots, b_n соответственно. $c(a_i)$ – $(n-1)$ -вершинный подграф, полученный из A удалением вершины a_i и всех смежных с ней ребер. Нам известно, что $c(a_i) \cong c(b_i)$.

Лемма 1. Каждый тип собственного подграфа, который встречается в A или B , встречается одинаковое количество раз в обоих, и a_i и b_i встречаются в одинаковом числе этих подграфов.

Лемма 2. Вершины a_i и b_i имеют одинаковую степень для $\forall i$.

Определение 2. Эксцентриситет v – наибольшее расстояние от v до любой другой вершины.

Центр графа – вершина с наименьшим эксцентриситетом.

Радиус графа – наименьший эксцентриситет. (Но в статье считаем, что радиус графа – наибольшее среди расстояний от вершины до ближайшего к ней центра)

Утверждение 1. У дерева может быть либо один, либо два центра. Такие деревья называются центральные и двуцентральные, соответственно.

Лемма 3. Деревья A и B имеют одинаковый радиус r , и одновременно являются центральными или двуцентральными.

Далее считаем, что оба дерева у нас двуцентральные.

Определение 3. r -точка – это такая точка, что расстояние от нее до ближайшего центра равно r .

Определение 4. a_i – несущественная точка, если $c(a_i)$ – остается двуцентральным графом с радиусом r .

Определение 5. Пусть \bar{a}_1 и \bar{a}_2 – центры дерева A . F – компонента в $A \setminus \{\bar{a}_1, \bar{a}_2\}$. Компонента F в A соединена ровно с одним центром (пусть с \bar{a}_1). Тогда граф $F \cup \bar{a}_1$ называется радиальным лимбом, если он содержит r -точку, а иначе нерадиальным.

A_r – совокупность всех радиальных лимбов в A .

Лемма 4. a_i – r -точка $\Leftrightarrow b_i$ – r -точка

Лемма 5. Если a_i – несущественная точка A в A_r , то b_i – несущественная точка B в B_r

Теорема 1. Если A и B – деревья с вершинами a_1, a_2, \dots, a_n и b_1, b_2, \dots, b_n и $c(a_i) \cong c(b_i)$ для $\forall i \Rightarrow A \cong B$

Доказательство состоит из разбора 3-х случаев:

1. Одно из деревьев имеет нерадиальный лимб
2. Нет нерадиальных лимбов, но одно из деревьев содержит хотя 3 радиальных лимба
3. У каждого дерева по два радиальных лимба

Современные методы в теоретической информатике

О связи между различными свойствами жёсткости

Золотов Б.

Основные определения

Определение: Конфигурация (G, \mathbf{p}) , состоящая из n точек и рёбер между ними, называется *локально жёсткой*, если не существует непрерывного движения точек конфигурации такого, что

- (а) оно сохраняет длины рёбер из G ,
- (б) оно **не** является сужением на точки $\mathbf{p}_1 \dots \mathbf{p}_n$ движения, состоящего из изометрий всего пространства \mathbb{R}^d .

Иными словами, мы могли, немного сдвинув точки, не изменить расстояние между ними, только если мы применили к ним глобальную изометрию, сохранив внешний вид конфигурации.

Определение: *Flex* конфигурации (G, \mathbf{p}) — это набор векторов $\mathbf{p}' = (\mathbf{p}'_1 \dots \mathbf{p}'_n)$ (всё вместе лежит в \mathbb{R}^{nd}) такой, что для любого ребра (i, j) из G верно

$$(\mathbf{p}_i - \mathbf{p}_j) \cdot (\mathbf{p}'_i - \mathbf{p}'_j) = 0.$$

Иными словами, это указание направлений в каждой вершине таких, что если начать двигать вершины в этих направлениях, то в нулевой момент времени при этом движении не будут растягиваться / сжиматься рёбра конфигурации.

Определение: Матрица жёсткости — это матрица $R(\mathbf{p})$ такая, что

$$R(\mathbf{p})\mathbf{p}' = \begin{pmatrix} \dots \\ (\mathbf{p}_i - \mathbf{p}_j) \cdot (\mathbf{p}'_i - \mathbf{p}'_j) \\ \dots \end{pmatrix}$$

Матрица имеет размер $e \times nd$ — будучи применена к вектору (\mathbf{p}'_k) размера nd , она даёт вектор высоты e , проиндексированный рёбрами $(i, j) \in G$. Заметим, что компоненты такой матрицы линейно зависят от координат вершин \mathbf{p} , поэтому её можно превратить в «билинейную форму»

$$R(\mathbf{p}, \mathbf{p}'): \quad \mathbb{R}^{nd} \times \mathbb{R}^{nd} \longrightarrow \mathbb{R}^e.$$

Таким образом, \mathbf{p}' — флекс для \mathbf{p} тогда и только тогда, когда

$$R(\mathbf{p}, \mathbf{p}') = 0.$$

Определение: Конфигурация (G, \mathbf{p}) называется *первопорядково жёсткой* (*infinitesimally rigid*), если у неё не существует нетривиальных флексов — таких, которые **не** получаются взятием в нуле первой производной гладкого движения из изометрий \mathbb{R}^d .

Определение: *Стресс* — сопоставление каждому ребру (i, j) числа $\omega_{ij} = \omega_{ji}$ (для тех пар вершин, которые не являются рёбрами, положим $\omega_{\dots} = 0$) такого, что для всякой вершины i

$$\sum_j \omega_{ij} \cdot (\mathbf{p}_i - \mathbf{p}_j) = 0.$$

Определение: Конфигурация (G, \mathbf{p}) называется *prestress stable*, если существует стресс ω такой, что для любого нетривиального флекса \mathbf{p}' выполнено

$$\sum_{i < j} \omega_{ij} \cdot (\mathbf{p}'_i - \mathbf{p}'_j)^2 > 0.$$

Иными словами, то влияние, которое \mathbf{p}' оказывает на длины рёбер конфигурации, незамедлительно портит сумму «весов», расставленных нами на рёбрах.

Определение: *Flex второго порядка* — два набора векторов $\mathbf{p}', \mathbf{p}''$ таких, что

$$\begin{aligned} R(\mathbf{p}, \mathbf{p}') &= 0; \\ R(\mathbf{p}, \mathbf{p}'') + R(\mathbf{p}', \mathbf{p}') &= 0. \end{aligned}$$

Определение: Конфигурация называется *второпорядково жёсткой*, если для неё не существует флексов второго порядка, у которых \mathbf{p}' нетривиален как флекс первого порядка.

Свойство, описанное выше, эквивалентно следующему ослаблению описанного ранее свойства prestress stability:

$$\forall \mathbf{p}' \text{ — нетривиального флекса } \exists \omega \text{ — стресс } \sum_{i < j} \omega_{ij} \cdot (\mathbf{p}'_i - \mathbf{p}'_j)^2 > 0.$$

Определение: *Привязанная жёсткость второго порядка* — то же самое, только ещё дано подмножество $G_0 \subset V(G)$, на котором \mathbf{p}' обязано обращаться в ноль.

Определение: Конфигурация называется *универсально *** жёсткой*, если соответствующее свойство выполнено при любой реализации данной конфигурации в произвольной размерности \mathbb{R}^D . Пример локально жёсткой, но не универсально локально жёсткой конфигурации — два треугольника с общей стороной.

Определение:

$$C'(d, G_0) = \left\{ \mathbf{p}' \in \mathbb{R}^{nd} \mid \mathbf{p}'_j = 0 \text{ на вершинах из } G_0 \right\}.$$

Супер-стабильность

Пусть дан стресс ω . Определим его энергию как

$$E_{\omega}(\mathbf{q}) = \sum_{i < j} \omega_{ij} (\mathbf{q}_i - \mathbf{q}_j)^2.$$

Понятно, что объект выше является квадратичной формой. Её матрица имеет вид $\mathbb{Id} : \otimes \Omega$. Ω называется *матрицей стресса*.

Определение: Конфигурация (G, \mathbf{p}) называется *супер-стабильной*, если для неё существует стресс ω , такой что

- (1) Матрица Ω для этого стресса положительно полуопределена;
- (2) Её ранг — $n - d - 1$, где d — размерность аффинной оболочки $\langle \mathbf{p} \rangle$;
- (3) Направления рёбер этой конфигурации не лежат на общей конике в проективном пространстве $P\langle \mathbf{p} \rangle$.

Теорема: Универсальная prestress stability равносильна супер-стабильности.

Классические результаты

Теорема: Всякий стресс $\omega \in \mathbb{R}^e$ лежит в коядре $R(\mathbf{p})$ — иными словами, перпендикулярен образу этого линейного отображения.

Теорема: Первопорядковая жёсткость \implies prestress stability \implies второпорядковая жёсткость \implies локальная жёсткость.

Теорема: Prestress stable-системы реализуют минимум энергии, определённой специальным образом.

Основные результаты

Определение: Если Y — выпуклый замкнутый конус, то *двойственный к нему* конус определяется как

$$Y^* = \{\omega \in \mathbb{R}^m \mid \langle \omega, y \rangle \geq 0, \forall y \in Y\}.$$

Теорема: Пусть Y — замкнутый выпуклый конус, L — линейное подпространство \mathbb{R}^m , пересекающееся с ним по нулю. Тогда существует элемент из ортогонального дополнения этого подпространства, лежащий во внутренней двойственного конуса Y^* .

Теорема: $Y = \{R(\mathbf{p}', \mathbf{p}') \mid \mathbf{p}' \in \mathcal{C}'(d, G_0)\} \subset \mathbb{R}^e$ — замкнутый выпуклый конус.

Теорема: Пусть (G, \mathbf{p}) — второпорядково жёсткая с привязанным множеством вершин G_0 , при этом аффинная оболочка у «привязанных» вершин — та же, что и у всех. Тогда Y пересекается разве что по нулю с линейным пространством L столбцов матрицы R .

Теорема: Любой вектор $\omega \in L^\perp$ является корректно определённым стрессом для (G, \mathbf{p}) . Любой вектор $\omega \in \text{Int}(Y^*)$ соответствует матрице Ω , положительно определённой на $\mathcal{C}'(1, G_0)$, подпространстве \mathbb{R}^n .

Теорема: Пусть (G, \mathbf{p}) — универсально второпорядково жёсткая конфигурация. Тогда у неё есть стресс с матрицей Ω , положительно определённой на $\mathcal{C}'(1, G_0)$.

Главный результат

Теорема: Пусть конфигурация (G, \mathbf{p}) — универсально второпорядково жёсткая, а также обладает d -мерной аффинной оболочкой. Тогда она обязана иметь стресс и матрицу стресса ранга $n - d - 1$, являющуюся положительно определённой — то есть, на самом деле, конфигурация должна быть супер-стабильной.

Отсюда универсальная супер-стабильность равносильна универсальной второпорядковой жёсткости.