Подсчет пересечений нормальных кривых на триангулированной поверхности

Иван Дынников

Математический институт им. В. А. Стеклова РАН

Декабрьские чтения в Томске — 2021

7 декабря 2021 г.

1. Что означает эффективно решить проблему равенства в группе с *практической* точки зрения?

1. Что означает эффективно решить проблему равенства в группе с *практической* точки зрения?
2. Каковы критерии эффективности?

1. Что означает эффективно решить проблему равенства в группе с *практической* точки зрения?
2. Каковы критерии эффективности?

Распространенная точка зрения:

1. Задать данную группу образующими и соотношениями и построить алгоритм, выясняющий, представляет ли произвольное данное слово единицу группы.

1. Что означает эффективно решить проблему равенства в группе с *практической* точки зрения?

2. Каковы критерии эффективности?

Распространенная точка зрения:

1. Задать данную группу образующими и соотношениями и построить алгоритм, выясняющий, представляет ли произвольное данное слово единицу группы.

2. Скорость этого алгоритма.

Пример

Группа: $\mathrm{SL}(2, \mathbb{Z})$

Образующие: $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$

Пример

Группа: $\mathrm{SL}(2, \mathbb{Z})$

Образующие: $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$, $\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$

Элементы $\begin{pmatrix} 5731478440138170841 & 3542248481792619075 \\ 3542248481792619075 & 2189229958345551690 26 \end{pmatrix}$ и $\begin{pmatrix} 1 & 100 \\ 0 & 1 \end{pmatrix}$ имеют одинаковую словарную длину.

Задание группы G:

- $\mathscr{A}$ — алфавит,

Задание группы G:

- $\mathscr{A}$ — алфавит,
- $\mathscr{L} \subset \mathscr{A}^*$ — подполугруппа (язык),

Задание группы G:

- $\mathscr{A}$ — алфавит,
- $\mathscr{L} \subset \mathscr{A}^*$ — подполугруппа (язык),
- $\pi : \mathscr{L} \to G$ — эпиморфизм.

Задание группы G:

- $\mathscr{A}$ — алфавит,
- $\mathscr{L} \subset \mathscr{A}^*$ — подполугруппа (язык),
- $\pi : \mathscr{L} \to G$ — эпиморфизм.

Пример: сжатая словарная запись

$a_1^{k_1} a_2^{k_2} \dots a_p^{k_p}$ будем кодировать как последовательность пар (a_1, k_1), (a_2, k_2), $\dots$, (a_p, k_p).

Для практических вычислений нужно:

- правое обратное к π (нормальная форма) $\mathrm{nf} : G \to \mathscr{L}$,

Для практических вычислений нужно:

- правое обратное к π (нормальная форма) $\mathrm{nf} : G \to \mathscr{L}$,
- алгоритм A_0 для проверки принадлежности $w \in \mathscr{L}$,

Для практических вычислений нужно:

- правое обратное к π (нормальная форма) $\mathrm{nf} : G \to \mathscr{L}$,
- алгоритм A_0 для проверки принадлежности $w \in \mathscr{L}$,
- алгоритм A_1 для вычисления $\mathrm{nf}(\pi(w))$,

Для практических вычислений нужно:

- правое обратное к π (нормальная форма) $\mathrm{nf} : G \to \mathscr{L}$,
- алгоритм A_0 для проверки принадлежности $w \in \mathscr{L}$,
- алгоритм A_1 для вычисления $\mathrm{nf}(\pi(w))$,
- алгоритм A_2 для вычисления $\mathrm{nf}(g^{-1})$, если известно $\mathrm{nf}(g)$.

Для практических вычислений нужно:

- правое обратное к π (нормальная форма) $\mathrm{nf} : G \to \mathscr{L}$,
- алгоритм A_0 для проверки принадлежности $w \in \mathscr{L}$,
- алгоритм A_1 для вычисления $\mathrm{nf}(\pi(w))$,
- алгоритм A_2 для вычисления $\mathrm{nf}(g^{-1})$, если известно $\mathrm{nf}(g)$.

Критерий эффективности:

скорость алгоритмов A_0, A_1 и A_2.
В частности, как быстро вычисляется $\mathrm{nf}(g_1 g_2)$, если даны $\mathrm{nf}(g_1)$ и $\mathrm{nf}(g_2)$.

Группа классов отображений $\mathrm{MCG}(M, \mathscr{P})$

M — компактная поверхность (достаточно сложная)
$\mathscr{P} \subset M$ — конечное непустое подмножество

$$\mathrm{MCG}(M, \mathscr{P}) = \mathrm{Diff}(M, \mathscr{P}) / \mathrm{Diff}_0(M, \mathscr{P})$$

Группа классов отображений $\mathrm{MCG}(M, \mathscr{P})$

M — компактная поверхность (достаточно сложная)
$\mathscr{P} \subset M$ — конечное непустое подмножество

$$\mathrm{MCG}(M, \mathscr{P}) = \mathrm{Diff}(M, \mathscr{P})/\mathrm{Diff}_0(M, \mathscr{P})$$

Решение проблемы равенства за квадратичное время по отношению к словарной метрике: Lee Mosher, 1995.

Кривая на M — открытая дуга в $M \setminus \mathscr{P}$ с «концами» в $\mathscr{P}$ или простая замкнутая кривая в $M \setminus \mathscr{P}$, в обоих случаях не отрезающая пустой диск.

Кривая на M — открытая дуга в $M \setminus \mathscr{P}$ с «концами» в $\mathscr{P}$ или простая замкнутая кривая в $M \setminus \mathscr{P}$, в обоих случаях не отрезающая пустой диск.
Мультикривая — объединение попарно не пересекающихся кривых.

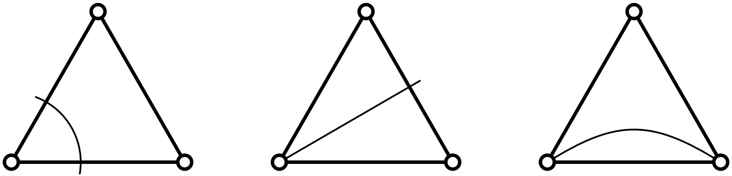
Кривая на M — открытая дуга в $M \setminus \mathscr{P}$ с «концами» в $\mathscr{P}$ или простая замкнутая кривая в $M \setminus \mathscr{P}$, в обоих случаях не отрезающая пустой диск.
Мультикривая — объединение попарно не пересекающихся кривых.
Геометрический индекс пересечения $\langle \gamma_1, \gamma_2 \rangle$ двух мультикривых — минимизированное с помощью изотопии относительно $\mathscr{P}$ число точек пересечения со следующими поправками:

- пересечение изотопных односторонних кривых не считается;
- параллельные дуги учитываются с весом -1.

Пусть $T = (e_1, \ldots, e_N)$ — триангуляция поверхности M с вершинами в $\mathscr{P}$ (в обобщенном смысле).

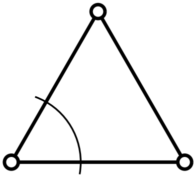
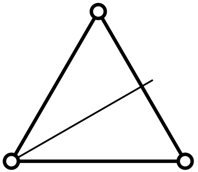
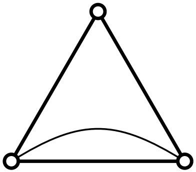
Пусть $T = (e_1, \ldots, e_N)$ — триангуляция поверхности M с вершинами в $\mathscr{P}$ (в обобщенном смысле).

Мультикривая γ *нормальна* по отношению к T, если пересекает каждый треугольник по *нормальным дугам*:



Пусть $T = (e_1, \dots, e_N)$ — триангуляция поверхности M с вершинами в $\mathscr{P}$ (в обобщенном смысле).

Мультикривая γ *нормальна* по отношению к T, если пересекает каждый треугольник по *нормальным дугам*:

Индексы $\langle e_i, \gamma \rangle$ — *нормальные координаты* мультикривой γ.

Матричное задание группы $\mathrm{MCG}(M, \mathscr{P})$

Элементу $g \in \mathrm{MCG}(M, \mathscr{P})$ сопоставим матрицу $m(g)$ следующим образом:

$$(m(g))_{ij} = \langle e_i, g \cdot e_j \rangle.$$

Будем иметь:

- если $g = g_1 g_2 \dots g_k$, то $|g| \leqslant \mathrm{const} \cdot (|g_1| + |g_2| + \ldots + |g_k|)$, где $|g|$ — длина записи матрицы $m(g)$;

Будем иметь:

- если $g = g_1 g_2 \ldots g_k$, то $|g| \leqslant \mathrm{const} \cdot (|g_1| + |g_2| + \ldots + |g_k|)$, где $|g|$ — длина записи матрицы $m(g)$;
- $m(g^{-1}) = m(g)^\top$;

Будем иметь:

- если $g = g_1 g_2 \ldots g_k$, то $|g| \leqslant \mathrm{const} \cdot (|g_1| + |g_2| + \ldots + |g_k|)$, где $|g|$ — длина записи матрицы $m(g)$;
- $m(g^{-1}) = m(g)^\top$;
- $m(g_1 g_2)_{ij}$ вычисляется по i-й строке матрицы $m(g_1)$ и j-му столбцу матрицы $m(g_2)$ как геометрический индекс пересечения мультикривых с такими нормальными координатами.

Основные результаты

- $m(g_1g_2)$ можно вычислить за время $O(|g_1| \cdot |g_2|)$;

Основные результаты

- $m(g_1 g_2)$ можно вычислить за время $O(|g_1| \cdot |g_2|)$;
- если $g = g_1 \dots g_k$, то $m(g)$ можно вычислить за квадратичное время по длине записи $g_1 \dots g_k$;

Основные результаты

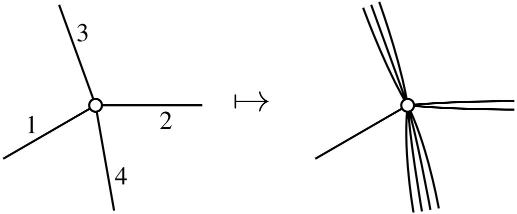
- $m(g_1 g_2)$ можно вычислить за время $O(|g_1| \cdot |g_2|)$;
- если $g = g_1 \ldots g_k$, то $m(g)$ можно вычислить за квадратичное время по длине записи $g_1 \ldots g_k$;
- сложность $|g|$ сравнима со сжатой словарной длиной, если образующие группы выбраны специальным образом.

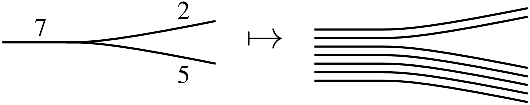
Основные результаты

- $m(g_1 g_2)$ можно вычислить за время $O(|g_1| \cdot |g_2|)$;
- если $g = g_1 \ldots g_k$, то $m(g)$ можно вычислить за квадратичное время по длине записи $g_1 \ldots g_k$;
- сложность $|g|$ сравнима со сжатой словарной длиной, если образующие группы выбраны специальным образом.

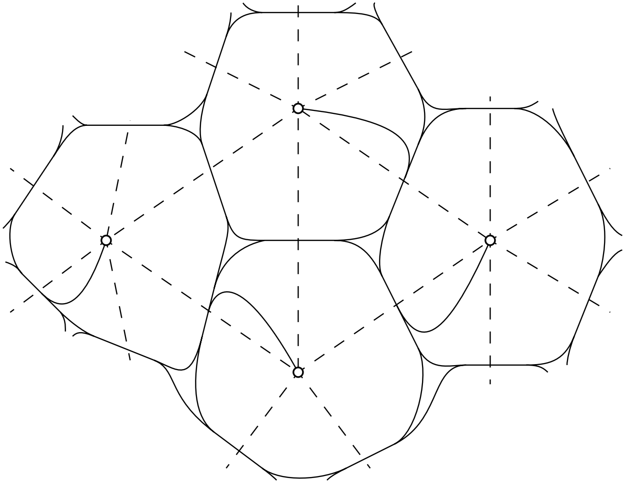
Метод доказательства: обобщение алгоритма Евклида (по аналогии с Agol–Hass–Thurston 2006) на основе одновременного расщепления трейнтреков.

Взвешенные трейнтреки

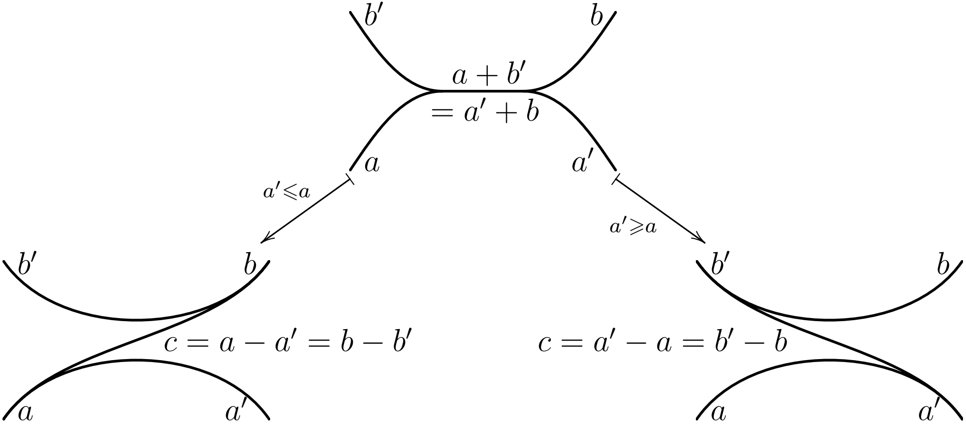




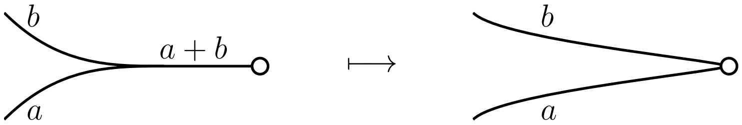
Универсальный трейнтрек



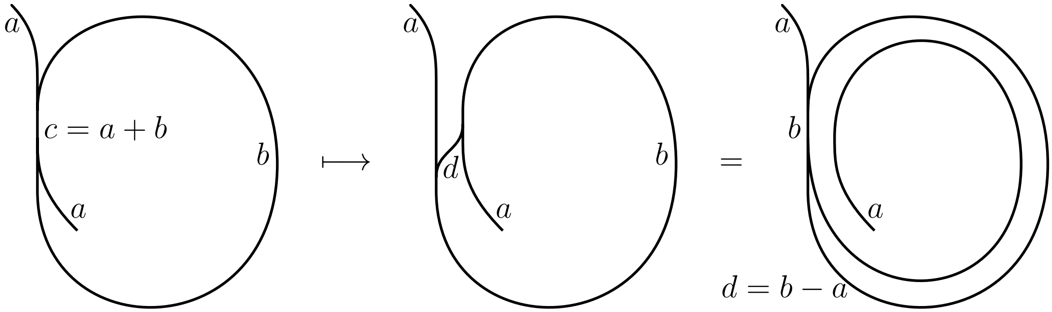
Расщепление



Расщепление

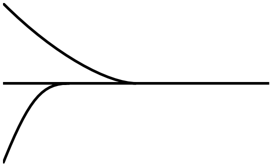


Кратное расщепление

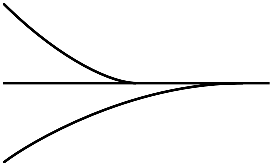


Делается одновременно $\left[\dfrac{a}{d}\right]$ раз.

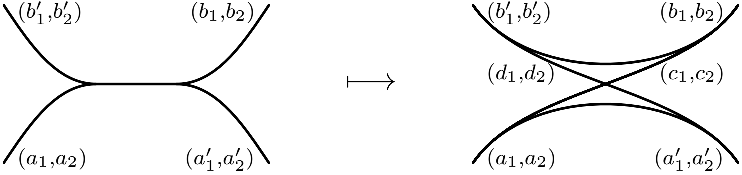
Скольжение



$$\longmapsto$$



Одновременное расщепление двух взвешенных трейнтреков



$$c_1 = \max(0, a_1 + b_1 - a'_1 - b'_1), \quad c_2 = \max(0, a_2 + b_2 - a'_2 - b'_2),$$
$$d_1 = \max(0, a'_1 + b'_1 - a_1 - b_1), \quad d_2 = \max(0, a'_2 + b'_2 - a_2 - b_2).$$

Бонус

В типичном случае возможно ускорение за счет быстрых алгоритмов умножения типа Карацубы.

Геометрический индекс пересечения можно непрерывно продолжить на пространство *ламинаций*.

Вопрос: что представляет собой пространство наборов ламинаций $(e_1, \ldots, e_N)$ со свойством $\langle e_i, e_j \rangle = -\delta_{ij}$?

Спасибо за внимание!