

УДК 519.85

Характеристики сложности: кликовое число графа многогранника и число прямоугольного покрытия

Максименко А. Н.¹

*Ярославский государственный университет им. П. Г. Демидова
150000 Россия, г. Ярославль, ул. Советская, 14*

e-mail: maximenko.a.n@gmail.com

получена 30 августа 2014

Ключевые слова: комбинаторная оптимизация, выпуклые многогранники, сложность задач и алгоритмов, граф многогранника, кликовое число, расширенные формулировки, матрица инциденций фасет-вершин, число прямоугольного покрытия

В 1980-х гг. В.А. Бондаренко обнаружил, что кликовое число графа многогранника во многих случаях соответствует реальной сложности задачи оптимизации на вершинах этого многогранника. Для объяснения этого феномена была предложена теория алгоритмов прямого типа, утверждающая, что кликовое число графа многогранника является нижней оценкой сложности соответствующей задачи в так называемом классе алгоритмов прямого типа. Более того, утверждалось, что этот класс является достаточно широким, включающим в себя многие классические комбинаторные алгоритмы. В настоящей работе приводятся несколько примеров, призванных обозначить границы применимости этой теории. В частности, описана довольно часто используемая на практике модификация алгоритмов, выводящая их из указанного класса (порядок трудоемкости при этом не меняется). Другой, значительно более близкой к реальности, комбинаторной характеристикой сложности является число прямоугольного покрытия матрицы инциденций фасет-вершин, введенное в рассмотрение М. Яннакакисом в 1988 г. Мы приводим пример многогранника с полиномиальным (относительно размерности многогранника) значением этой характеристики, задача оптимизации на вершинах которого NP-трудна.

Введение

Рассмотрим следующую задачу квадратичного 0-1 программирования.

Задан многочлен второй степени от n переменных:

$$P(x) = P(x_1, \dots, x_n) = \sum_{1 \leq i \leq j \leq n} c_{ij} x_i x_j, \quad x = (x_1, \dots, x_n) \in \mathbb{R}^n.$$

¹Работа выполнена при поддержке проекта №477 в рамках базовой части государственного задания на НИР ЯрГУ.

Точнее, задан вектор его коэффициентов $c \in \mathbb{R}^{n(n+1)/2}$ с координатами $c_{ij} \in \mathbb{R}$.

ТРЕБУЕТСЯ найти точку $x \in \{0, 1\}^n$, в которой многочлен достигает своего максимума.

Отметим, что эта задача имеет простую формулировку, но при этом является NP-трудной [4]. По-видимому, сочетание этих двух фактов и обуславливает повышенный интерес к исследованию ее свойств. Особенно продуктивными в последнее время оказались исследования свойств ассоциированного с этой задачей *булева квадратичного многогранника*, определяемого как выпуклая оболочка

$$\text{BQR}_n = \text{conv} \{x = (x_{ij}) \in \{0, 1\}^{n(n+1)/2} \mid x_{ij} = x_{ii}x_{jj}, 1 \leq i < j \leq n\}.$$

Соответственно, задача квадратичного 0-1 программирования может быть сформулирована как задача оптимизации $\langle c, x \rangle \rightarrow \max, x \in \text{BQR}_n$. Иная постановка задачи открывает новые подходы к ее решению. Тем не менее, несмотря на значительное продвижение в этом направлении, проблема труднорешаемости до сих пор не снята. Т.е. многочисленные результаты в этой области по большому счету носят отрицательный характер. В частности, число фасет (гиперграней) многогранника BQR_n суперэкспоненциально по параметру n [12]. Более того, известно [13] (см. также [17]), что если BQR_n является аффинным образом некоторого многогранника P (в таком случае P называется *расширенной формулировкой*² многогранника BQR_n), то число фасет этого многогранника P не может быть меньше $2^{n/2}$. Известно также, что BQR_n является 3-смежностным³ многогранником [12]. Более того, для любого $k \leq 3 \lfloor \frac{\log_2 n}{2} \rfloor$, BQR_n имеет k -смежностную грань с суперполиномиальным числом $2^{\Theta(n^{1/\lceil k/3 \rceil})}$ вершин [8].

Пользуясь аналогичными соображениями, определяются многогранники многих других задач комбинаторной оптимизации. В качестве еще одного примера, используемого далее, приведем определение многогранника совершенных паросочетаний в полном графе. Для каждого совершенного паросочетания M в полном графе $G = G(V, E)$ на четном числе вершин $V = \{1, \dots, n\}$ рассмотрим его характеристический вектор $x = \chi(M) \in \mathbb{R}^E$ с координатами

$$x_{ij} = \begin{cases} 1, & \text{если ребро } \{i, j\} \text{ входит в } M, \\ 0, & \text{иначе.} \end{cases}$$

Многогранником совершенных паросочетаний называется выпуклая оболочка всех таких векторов:

$$\text{MATSN}_n = \text{conv} \{\chi(M) \mid M - \text{совершенное паросочетание в } G\}.$$

Этот многогранник имеет размерность $n(n-3)/2$ и экспоненциальное число вершин и фасет. Кроме того, совсем недавно было показано [19], что если MATSN_n является аффинным образом некоторого многогранника P (P является расширенной формулировкой MATSN_n), то число фасет многогранника P экспоненциально по n .

²Размерность расширенной формулировки обычно значительно больше размерности исходного многогранника.

³Любые k вершин k -смежностного многогранника образуют его грань ($(k-1)$ -мерный симплекс).

Обратим внимание на то, что при переходе к многограннику формулировка задачи приобретает четкий структурированный вид. Заметим также, что эта структура неявно присутствует в задаче и без определения ее многогранника. Полиэдральная формулировка лишь делает ее более явной. Чтобы уточнить эту мысль, рассмотрим произвольную задачу дискретной оптимизации следующего вида.

ЗАДАНО некоторое (конечное) множество допустимых решений X , вектор входных данных $c \in \mathbb{R}^d$ и некоторая скалярная функция $f = f(c, x)$, $x \in X$.

ТРЕБУЕТСЯ найти решение $x \in X$, при котором функция $f = f(c, x)$ достигает своего максимального (или минимального) значения.

Прежде всего заметим, что из множества X можно выбросить решения, которые не являются (единственными) оптимальными ни при каком входе $c \in \mathbb{R}^d$. Те решения, которые останутся, соответствуют вершинам многогранника задачи. (Разумеется, о многограннике мы можем говорить лишь при условии, что функция f линейна, а $X \subset \mathbb{R}^d$.) Следуя [1], два решения $x, y \in X$ будем называть *смежными*, если найдется такой вход $c \in \mathbb{R}^d$, что

$$f(c, x) = f(c, y) > f(c, z), \quad \forall z \in X, z \neq x, z \neq y. \quad (1)$$

При переходе к многограннику задачи это означает, что вершины x и y соединены ребром (1-гранью) многогранника. Аналогичным образом можно определить подмножества решений, образующих грани более высоких размерностей.

Таким образом, переход к многограннику делает понятие смежных решений (а также граней более высоких размерностей) более «выпуклым». При этом, разумеется, задача приобретает также некоторые свойства геометрического (некомбинаторного) характера, которые, по-видимому, являются не менее важными, но в контексте настоящей работы отходят на второй план. Далее мы будем рассматривать только комбинаторные свойства многогранника, полностью определяемые его решеткой граней⁴, и попробуем частично ответить на следующий вопрос: верно ли, что существует числовая характеристика, определяемая комбинаторной структурой задачи, адекватно отражающая вычислительную сложность этой задачи? Точнее, в настоящей работе мы снимем этот вопрос в отношении нескольких таких характеристик.

Начнем с простых, естественных примеров характеристик:

1. Собственная размерность многогранника служит нижней оценкой трудоемкости для соответствующей задачи. (Заметим, что собственная размерность многогранника может быть значительно меньше размера входных данных задачи.)
2. Число вершин многогранника, очевидно, является верхней оценкой трудоемкости.
3. Число фасет многогранника (точнее, некоторый полином от этого числа) также является верхней оценкой трудоемкости соответствующей задачи (хотя бы потому, что задача линейного программирования полиномиально разрешима).

⁴Множество граней (всех размерностей) многогранника, упорядоченное по включению, называется решеткой граней.

4. Долгое время в качестве претендента на характеристику сложности задачи рассматривался диаметр графа (1-скелета) ее многогранника. Несостоятельность этой оценки однозначно доказывается тем фактом, что диаметр графа многогранника BQP_n равен 1, тогда как задача квадратичного 0-1 программирования NP-трудна.

Заметим, что размерность многогранника, число его вершин и число фасет могут служить лишь (часто весьма далекими от реальности) оценками трудоемкости соответствующей задачи комбинаторной оптимизации. В качестве примера можно привести многогранник $MATCH_n$. Его размерность квадратична по основному параметру n , число вершин и число фасет экспоненциальны. В то же время лучшие из современных алгоритмов для решения этой задачи используют порядка n^3 операций. Заметим для сравнения, что размерность многогранника BQP_n имеет тот же порядок, его число вершин и число фасет экспоненциальны, но соответствующая задача относится к классу NP-трудных, т.е. полиномиальные по трудоемкости алгоритмы для нее не известны. Как видно, оценка снизу (размерность многогранника) и оценки сверху (число вершин и число фасет) во многих случаях⁵ не дают возможности судить о реальной сложности задач.

Ниже рассматриваются три существенно более интересных примера такого рода характеристик. Это кликовое число графа многогранника, сложность расширенной формулировки многогранника и ее комбинаторный аналог (число прямоугольного покрытия матрицы инциденций фасет-вершин). В каждом случае удается построить примеры задач, реальная вычислительная сложность которых существенно отличается от соответствующих значений характеристик.

1. Кликовое число полиэдрального графа и алгоритмы прямого типа

В монографии [1] кликовое число графа многогранника рассматривается как нижняя оценка сложности соответствующей задачи в «некотором широком» классе так называемых алгоритмов прямого типа. Алгоритмы прямого типа относятся к классу линейных разделяющих алгоритмов, которые удобно представлять в виде линейных разделяющих деревьев.

Далее в этом разделе вместе с каждым допустимым решением x задачи $X \subset R^m$ рассматривается конус входных данных

$$K(x) = \{c \in R^m \mid \langle x, c \rangle \geq \langle y, c \rangle \forall y \in X\}, \quad (2)$$

для которых этот x является оптимальным (или одним из оптимальных) решением.

Определение 1. [1, с. 33]. Линейным разделяющим деревом задачи X , $X \subset R^m$, называется ориентированное дерево, обладающее следующими свойствами:

- а) в каждый узел, за исключением одного, называемого корнем, входит ровно одна дуга; дуг, входящих в корень, нет;

⁵Приведенные примеры типичны для комбинаторной оптимизации.

- б) для каждого узла либо имеется две выходящих из него дуги, либо таких дуг нет вообще; в первом случае узел называется внутренним, во втором — внешним, или листом;
- в) каждому внутреннему листу соответствует некоторый линейный функционал (вектор) f в R^m ;
- г) каждому листу соответствует некоторый элемент множества X ;
- д) каждой дуге d соответствует число $\text{sgn } d$, равное 1 либо -1 ; две дуги, выходящие из одного узла, имеют различные значения;
- е) для каждой цепи $W = f_1 d_1 f_2 d_2 \dots f_l d_l x$, соединяющей корень и лист (в обозначении цепи перечислены соответствующие ее узлам векторы; дуга d_i выходит из узла f_i , $i = 1, \dots, l$), и для любого $c \in R^m$ из неравенств $\langle f_i, c \rangle \text{sgn } d_i \geq 0$, $i = 1, \dots, l$, следует включение $c \in K(x)$.

Сложностью $C_{LSA}(X)$ задачи X в классе линейных разделяющих алгоритмов называется высота (глубина) минимального линейного разделяющего дерева этой задачи. Очевидно, $C_{LSA}(X) \geq \log_2 |X|$ (разумеется, при условии, что X не содержит решений, не являющихся оптимальными ни при каком входе c). В 1982 г. М.Ю. Мошков [9] показал, что $C_{LSA}(X) = O(m^3 \log_2 |X|)$ для любой задачи $X \in \mathbb{R}^m$. То есть NP-трудные задачи приравниваются к полиномиально разрешимым в классе линейных разделяющих алгоритмов. Такое положение дел заставляет предположить, что существующие алгоритмы удовлетворяют некоторым дополнительным ограничениям, существенно снижающим их эффективность. Одно из таких ограничений было предложено в 1980-х гг. В.А. Бондаренко.

Введем, следуя [1], несколько вспомогательных обозначений. Пусть T — линейное разделяющее дерево задачи X и пусть f — его внутренний узел. Обозначим через X_f множество пометок всех листьев дерева T , которым предшествует узел f , а через X_f^+ и X_f^- обозначим подмножества X_f , соответствующие двум выходящим из f дугам. Обозначим через $R_f^- = X_f^+ \setminus X_f^-$ множество пометок, отбрасываемых при переходе по «отрицательной» дуге. По аналогии определим множество пометок $R_f^+ = X_f^- \setminus X_f^+$, отбрасываемых при переходе по «положительной» дуге.

В [1] предложены два формально разных варианта определения алгоритма прямого типа, в основе которых лежит одна и та же идея, для описания которой понадобится понятие клики решений. Множество решений $Y \subseteq X$ называется *кликой*, если любые два решения $x, y \in Y$ смежны (т.е. при некотором входе c удовлетворяют неравенству (1)). Мощность максимальной клики в X обозначаем $\omega(X)$.

Определение 2. [1, с. 39]. Линейное разделяющее дерево T задачи X называется деревом прямого типа, если для любого внутреннего узла f и для любой клики $Y \subseteq X$ выполняется неравенство

$$\min\{|R_f^+ \cap Y|, |R_f^- \cap Y|\} \leq 1. \quad (3)$$

Определение 3. [1, с. 40]. Деревом «прямого типа» задачи X , $X \in R^m$, называется линейное разделяющее дерево этой задачи, для которого каждая цепь $w = f_1 d_1 f_2 d_2 \dots f_l d_l x$, соединяющая корень и лист, удовлетворяет условиям:

(*) для любого $y \in X$ смежного с x , найдется такой номер i , $1 \leq i \leq l$, что условия $\langle f_i, c \rangle \operatorname{sgn} d_i > 0$ и $c \in K(y)$ несовместны;

(**) для любого $i = 1, 2, \dots, l$ из несовместности условий

$$\langle f_i, c \rangle \operatorname{sgn} d_i > 0 \quad \text{и} \quad c \in K(y)$$

для y , смежного с x , и из телесности конуса

$$K(x) \cap \{c \in R^m : \langle f_i, c \rangle \operatorname{sgn} d_i \leq 0\}$$

следует, что ветвь, начинающаяся в узле f_i с дугой $-d_i$, имеет хотя бы один лист, помеченный x .

Оба этих определения объединяет следующий факт.

Теорема 1. [1, раздел 2.4]. *Высота дерева прямого типа («прямого типа») задачи X не меньше $\omega(X) - 1$.*

Таким образом, мощность максимальной клики является нижней оценкой трудоемкости для такого типа алгоритмов. Как показали многочисленные исследования, для классических полиномиально разрешимых задач (сортировка, минимальное остовное дерево, кратчайший путь в графе, минимальный разрез) эта характеристика не превосходит размерности многогранника [2, 3]. С другой стороны, оказалось, что BQR_n является гранью многогранников таких труднорешаемых задач, как коммивояжер, рюкзак, 3-выполнимость, 3-сочетание, покрытие и упаковка множества, кубический подграф и многие другие [18]. Учитывая, что кликовое число графа многогранника BQR_n равно 2^n , кликовые числа графов многогранников указанных задач также суперполиномиальны по размерности многогранников. Кроме того, в [1] установлено, что некоторые алгоритмы сортировки, жадные алгоритмы для минимального остовного дерева, алгоритм Дейкстры для кратчайшего пути, алгоритм Хелда–Карпа и реализация алгоритма ветвей и границ для задачи коммивояжера являются прямыми или «прямыми». (Хотя, в контексте приводимых ниже замечаний, некоторые из этих результатов могут быть подвергнуты сомнению.)

Ограниченность такого подхода к оценке сложности задач иллюстрируется следующими фактами.

Пример 1. Как уже было сказано выше, для многогранников полиномиально разрешимых задач кликовое число графа, как правило, не превосходит размерности многогранника. Т.е. такая тривиальная оценка, как размерность многогранника, мажорирует эту характеристику. Кроме того, явно сравнимые по сложности задачи могут иметь «несравнимые» кликовые числа. Например, задача выбора наибольшего элемента в массиве из n элементов очевидно проще задачи сортировки. Многогранник первой задачи представляет собой симплекс [1]. Его граф полный. Следовательно, кликовое число равно n . Многогранник второй задачи называется *перестановочным*, а кликовое число его графа равно 2 [15]. Т.е. в данном случае кликовое число никак не связано с реальной сложностью задачи.

Пример 2. В [1] на основе релаксации многогранника BQR_n строится пример полиномиально разрешимой задачи с экспоненциальным кликовым числом графа многогранника.

Пример 3. Для любых натуральных k и m известны примеры k -смежных многогранников на m вершинах, допускающих полиномиальное относительно $\log m$ описание [10].

Приведенные примеры говорят о том, что для полиномиально разрешимых задач размерность многогранника во многих случаях оказывается гораздо более точной характеристикой сложности, чем кликовое число графа. В случае же многогранников NP-трудных задач высокое значение кликового числа объясняется тем, что многогранник VQR_n оказывается гранью этих многогранников в силу естественных причин, обусловленных (аффинной) сводимостью [18].

В связи с этим естественным оказывается следующий вопрос: насколько широким является класс алгоритмов прямого («прямого») типа? Отметим, что ранее в [6] уже приводился пример достаточно простого алгоритма для решения задачи о назначениях, который не является прямым. В настоящей работе будет показано, что алгоритм Куна–Манкреса (известный также под названием венгерского алгоритма) не является алгоритмом прямого (а также «прямого») типа. Кроме того, будет описан достаточно универсальный способ модификации алгоритмов, существенно не меняющий их трудоемкости, но гарантированно выводящий их из указанных классов.

Теорема 2. Алгоритм Куна–Манкреса для задачи о назначениях не является алгоритмом прямого (а также «прямого») типа.

Доказательство. Далее предполагаем, что в рассматриваемой задаче о назначениях требуется в заданном реберно взвешенном полном двудольном графе найти совершенное паросочетание минимального веса. Набор весов ребер является вектором входных данных задачи. Его удобно представлять в виде матрицы

$$c = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{pmatrix},$$

где c_{ij} — вес ребра, соединяющего i -ю вершину «нижней» доли графа и j -ю вершину «верхней» доли. При таком подходе допустимые решения задачи представляются в виде булевых $n \times n$ -матриц, каждая строка и каждый столбец которых содержат ровно по одной единице. Выпуклая оболочка всех таких матриц называется *многогранником Биркгофа*.

Для доказательства нам не потребуется описание всего алгоритма Куна–Манкреса. Достаточно рассмотреть этап предобработки (редуцирования строк и столбцов матрицы весов). На этом этапе перебираются сначала вершины одной доли графа (например, «нижней»), затем другой («верхней»). Для каждой вершины среди инцидентных ей ребер выбирается ребро с наименьшим весом. Его вес вычитается из весов всех инцидентных этой вершине ребер. Ребро с наименьшим весом при этом приобретает нулевой вес, все остальные — неотрицательный. Заметим, что после прохождения этого этапа возможна ситуация, когда ребра с нулевым весом образуют паросочетание. Тогда, разумеется, именно оно окажется минимальным, и задача решена.

Особо следует обратить внимание на то, что на этом этапе обычно не оговаривается способ выбора ребра с минимальным весом. Без явного описания этого способа рассуждения о принадлежности к классу алгоритмов прямого типа становятся беспредметными. Обычно предполагается следующий естественный способ (к тому же являющийся оптимальным по трудоемкости). На первом шаге сравниваются веса двух произвольно выбранных ребер. Не уменьшая общности, можно считать, что это первое и второе ребро (в данной выборке). Далее, наименьшее из них сравнивается с весом третьего ребра и т.д. В результате последнего $((n-1)$ -го, если элементов n) сравнения выявляется ребро с наименьшим весом.

1.1. Алгоритм Куна–Манкреса не является алгоритмом прямого типа.

Согласно определению 2 достаточно указать внутренний узел f (точнее, некоторую операцию сравнения) в дереве алгоритма и клику из четырех допустимых решений Y так, чтобы при любом результате сравнения два из этих четырех решений оказывались бы отброшенными:

$$|R_f^+ \cap Y| = |R_f^- \cap Y| = 2.$$

Ниже будет приведен пример для $n = 4$ вершин в каждой доле, который легко масштабируется на произвольные значения $n > 4$.

Рассмотрим полный двудольный граф с четырьмя вершинами в каждой доле. Набор весов ребер представляем в виде матрицы

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{pmatrix}$$

Предположим, как было сказано ранее, что для каждой строки, начиная с первой, выполняется следующая процедура. Вес c_{i1} сравнивается с весом c_{i2} . Затем наименьший из них сравнивается с c_{i3} . И, наконец, наименьший из последних двух сравнивается с c_{i4} .

Обратим внимание на то, что случаи равенства сравниваемых весов для теории алгоритмов прямого типа не информативны. В противном случае ни в одном узле соответствующего линейного разделяющего дерева не отбрасывалось бы ни одно решение. (Это рассуждение носит общий характер и распространяется не только на данный алгоритм.)

Предположим теперь, что мы прошли по ветке линейного разделяющего дерева со следующими дугами (дуга представляет собой результат сравнения пары весов): 1) $c_{11} < c_{12}$; 2) $c_{11} < c_{13}$; 3) $c_{11} < c_{14}$; 4) $c_{21} > c_{22}$; 5) $c_{22} < c_{23}$; 6) $c_{22} < c_{24}$.

В настоящий момент мы находимся в узле $c_{31} < c_{32}$, а матрица приобрела следующий вид:

$$\begin{pmatrix} 0 & c'_{12} & c'_{13} & c'_{14} \\ c'_{21} & 0 & c'_{23} & c'_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{pmatrix},$$

где $c'_{1i} = c_{1i} - c_{11}$, $c'_{2i} = c_{2i} - c_{22}$, $i = 1, 2, 3, 4$. Заметим, что все «не нулевые» веса в этой матрице могут принимать любые положительные значения, никак не

зависящие друг от друга. (Не уменьшая общности, можно считать, что все исходные веса положительны.)

Не трудно проверить, что среди всех 24 паросочетаний данного графа отброшенными по итогам шести сравнений окажутся лишь следующие два (для удобства перечислим обозначения весов соответствующих ребер): $\{c'_{12}, c'_{21}, c_{33}, c_{44}\}$ и $\{c'_{12}, c'_{21}, c_{34}, c_{43}\}$. Т.е. множество X_f узла $c_{31} < c_{32}$ состоит из 22 паросочетаний.

Рассмотрим теперь результаты сравнения $c_{31} < c_{32}$.

1) Если верно $c_{31} < c_{32}$, то отброшены будут паросочетания $\{c'_{21}, c_{32}, c'_{13}, c_{44}\}$ и $\{c'_{21}, c_{32}, c'_{14}, c_{43}\}$.

2) Если верно $c_{31} > c_{32}$, то отброшены $\{c'_{12}, c_{31}, c'_{23}, c_{44}\}$ и $\{c'_{12}, c_{31}, c'_{24}, c_{43}\}$.

Причем все четыре указанных паросочетания попарно смежны. Это легко проверяется, например, с помощью известного критерия смежности вершин многогранника Биркгофа (см., например, [5]).

Лемма 1 (критерий смежности). *Два паросочетания смежны тогда и только тогда, когда объединение их ребер содержит ровно один цикл.*

Следовательно, при любом исходе сравнения $c_{31} < c_{32}$ от данной клики из четырех решений отбрасывается ровно два.

1.2. Алгоритм Куна–Манкреса не является алгоритмом «прямого» типа.

Как было сказано ранее, уже после прохождения этапа «предобработки» возможна ситуация, когда ребра с нулевым весом образуют паросочетание. В такой ситуации мы попадаем в лист линейного разделяющего дерева. При этом цепь, соединяющая корень дерева и этот лист, состоит только из узлов вида $c_{ij} < c_{il}$.

Заметим, что для любого паросочетания можно подобрать веса ребер так, что наперед заданное неравенство $c_{ij} < c_{il}$ будет выполнено, а суммарный вес ребер этого паросочетания окажется меньше веса любого другого паросочетания. Ровно то же самое справедливо и в отношении неравенства $c_{ij} > c_{il}$. Другими словами, каждый конус $K(x)$ (см. формулу (2)) имеет общие внутренние точки с каждым из полупространств $c_{ij} < c_{il}$ и $c_{ij} > c_{il}$. Следовательно, условие (*) из определения 3 нарушается для «коротких» цепей, заканчивающихся на этапе предобработки. □

В [1, с. 77] приводится «доказательство» того, что алгоритм Эдмондса для задачи о паросочетаниях в произвольном реберно взвешенном графе является алгоритмом прямого типа. (Позднее этот факт дублируется в [2].) Слово «доказательство» здесь намеренно взято в кавычки, так как сам алгоритм Эдмондса представляет собой достаточно непростую модификацию алгоритма Куна–Манкреса, а анализ его принадлежности к алгоритмам прямого типа в [1] занимает полстраницы, где по существу излагается лишь идея доказательства, оставляющая читателю широкие возможности для самостоятельных исследований. Учитывая, что основой алгоритма Эдмондса является алгоритм Куна–Манкреса, доказательство теоремы 2 переносится и на этот случай.

Следствие 1. *Алгоритм Эдмондса для задачи о паросочетаниях в произвольном графе не является алгоритмом прямого (а также «прямого») типа.*

При внимательном рассмотрении полученный результат говорит о том, что теория алгоритмов прямого типа в форме определений 2 и 3 чувствительна к этапу предобработки, на котором отсеиваются простые случаи. Развивая эту идею, приведем еще один пример непрямого алгоритма.

Рассмотрим задачу о кратчайшем пути.

ЗАДАН полный реберно взвешенный граф на n вершинах, в котором выделены две вершины s и t .

ТРЕБУЕТСЯ найти в этом графе простой путь минимальной длины (точнее, с минимальным суммарным весом ребер), соединяющий вершины s и t .

Хорошо известно [4], что при отсутствии ограничений на веса ребер эта задача является NP-трудной. Более того, ее многогранник содержит в качестве грани многогранник задачи коммивояжера, а следовательно, кликовое число графа многогранника задачи о кратчайшем пути суперполиномиально [7]. С другой стороны, при ограничении неотрицательности длин ребер задача становится полиномиально разрешимой (ее трудоемкость сопоставима с числом ребер в исходном графе). Воспользуемся этим, чтобы построить пример непрямого алгоритма.

Пример 4. Рассмотрим какой-нибудь алгоритм (прямого типа) для решения задачи о кратчайшем пути в общем виде. Добавим к нему этап предобработки, обслуживающий те частные случаи, когда выполняется условие неотрицательности длин ребер. (Заметим, что эта идея довольно часто реализуется на практике, так как помогает существенно ускорить работу алгоритма для простых случаев.) Соответственно, общая глубина дерева алгоритма увеличится на число операций, необходимых для проверки условия неотрицательности. Особо обратим внимание на то, что в ходе этой проверки ни одно решение не отбрасывается. (Хотя бы потому, что длины ребер сравниваются с нулем, а не друг с другом.) В том месте (узле), где вся эта проверка успешно заканчивается, у дерева появляется еще одна ветвь, реализующая решение для случая неотрицательных длин. Причем множество пометок листьев этой ветви совпадает с множеством всех решений, (т.к. ни одно решение в ходе проверки условия неотрицательности не было отброшено). Важно то, что эта ветвь полностью сохраняет множество решений исходной общей задачи, но при этом имеет полиномиальную глубину, так как соответствующий частный случай полиномиально разрешим. С другой стороны, как было замечено ранее, кликовое число для этой задачи суперполиномиально. Сопоставляя эти факты с теоремой 1, приходим к выводу, что данный алгоритм не является ни прямым, ни «прямым».

Все это говорит о том, что ограничения, накладываемые определениями 2 и 3, не являются естественными. Кроме того, проверка этих ограничений (в том случае, если они действительно выполняются), как правило, сложнее непосредственной оценки трудоемкости алгоритма, что существенно снижает ценность теоремы 1.

В заключение заметим, что, несмотря на указанные недостатки, теория алгоритмов прямого типа послужила отправной точкой для ряда новых исследований, открывающих завесу над глубинными причинами труднорешаемости задач комбинаторной оптимизации. В частности, автор настоящей работы благодарен В.А. Бондаренко за знакомство с этой интересной теорией.

2. Расширенные формулировки и число прямоугольного покрытия

Многогранник $Q \subset \mathbb{R}^m$ называется *расширением* (extension) многогранника $P \subset \mathbb{R}^n$, если существует (как правило, вырожденное) аффинное (линейное) отображение $\pi : \mathbb{R}^m \rightarrow \mathbb{R}^n$ такое, что $P = \pi(Q)$. Любое описание многогранника Q с помощью линейных уравнений и неравенств (вместе с π) называется *расширенной формулировкой* (extended formulation) многогранника P . *Размером* расширенной формулировки (расширения) называется число линейных неравенств в ее описании (число фасет расширения). *Сложностью расширенной формулировки* $xc(P)$ многогранника P называется минимальный размер его расширения. Интерес к расширенным формулировкам вызван в первую очередь тем, что в некоторых случаях размер расширенной формулировки оказывается значительно меньше описания исходного многогранника. Для первого знакомства с расширенными формулировками можно воспользоваться [16]. Более подробный обзор содержится в [11]. В последние годы в этой области был получен ряд прорывных результатов. Упомянем лишь два из них, имеющие важное значение в контексте настоящей работы. В 2012 году было показано, что сложность расширенной формулировки для VQR_n экспоненциальна по n (суперполиномиальна по размерности многогранника) [13]. (Чтобы подчеркнуть масштабность этого результата, заметим, что VQR_n является гранью многих других многогранников, ассоциированных с труднорешаемыми задачами, точнее, аффинно сводится к ним [18]. Откуда следует, что сложность расширенных формулировок этих многогранников также суперполиномиальна по размерности.) В 2014 году Т. Ротвоссу удалось продвинуться еще дальше. Он доказал, что то же самое утверждение верно и для многогранника паросочетаний $MATCH_n$ [19]. Этот в некоторой мере неожиданный результат говорит о том, что сложность расширенной формулировки не всегда соответствует реальной сложности задачи. Тем не менее, эта характеристика оказывается существенно ближе к реальности, чем кликовое число графа многогранника. Так, например, сложность расширенной формулировки 4-мерного циклического многогранника на n вершинах не превышает $O(\log^2 n)$ [10], тогда как кликовое число его равно n .

Еще более близким к реальности претендентом на роль характеристики сложности является комбинаторный суррогат сложности расширенной формулировки. Для его описания нам потребуется матрица инцидентий фасет и вершин многогранника и понятие ее покрытия 0-прямоугольниками. Пусть m — число фасет многогранника P , а n — число его вершин, тогда *матрица инцидентий фасет-вершин* $A = (a_{ij}) \in \{0, 1\}^{m \times n}$ определяется следующим образом: $a_{ij} = 1$, если j -я вершина принадлежит i -й фасете и $a_{ij} = 0$ в остальных случаях. *Прямоугольником* будем называть множество $I \times J$, где I и J — подмножества индексов строк и столбцов соответственно. Прямоугольник $I \times J$ называется *0-прямоугольником*⁶ в матрице A , если $a_{ij} = 0$ для любой пары $(i, j) \in I \times J$. Множество 0-прямоугольников называется *прямоугольным покрытием* матрицы инцидентий A , если каждый нулевой элемент матрицы принадлежит хотя бы одному прямоугольнику из этого множества. *Размером* прямоугольного покрытия будем называть его мощность (число

⁶Обычно в контексте расширенных формулировок рассматривается матрица неинцидентий и тогда речь идет об 1-прямоугольниках.

прямоугольников). *Числом прямоугольного покрытия* (rectangle covering number) $rc(A)$ матрицы A называется минимальный размер ее прямоугольного покрытия.

Между числом прямоугольного покрытия $rc(A)$ матрицы инцидентий фасет-вершин многогранника P и сложностью его расширенной формулировки $xc(P)$ имеется следующая фундаментальная взаимосвязь [20]:

$$rc(A) \leq xc(P).$$

В частности, экспоненциальная нижняя оценка для $xc(\text{BQR}_n)$ является следствием экспоненциальной нижней оценки для числа прямоугольного покрытия матрицы инцидентий этого многогранника [13]. Так как семейство булевых квадратичных многогранников аффинно сводится к многим другим семействам многогранников, ассоциированных с труднорешаемыми задачами (в их числе задача коммивояжера, задача о рюкзаке, 3-выполнимость, 3-сочетание и многие другие), то они наследуют эту характеристику сложности [18]. С другой стороны, число прямоугольного покрытия весьма точно отражает реальную сложность полиномиально разрешимых задач. Во всяком случае, список задач, для которых этот факт установлен, весьма внушителен [11]. В частности, для многогранника паросочетаний MATCH_n число прямоугольного покрытия больше, чем n^2 , но меньше, чем n^4 [14]. (Напомним, что лучшие из известных алгоритмов решают задачу о паросочетаниях за $O(n^3)$ операций.) В то время как $xc(\text{MATCH}_n)$ экспоненциальна по n . Все это наталкивает на мысль о том, что это и есть та самая идеальная характеристика сложности соответствующей задачи комбинаторной оптимизации... Оказывается, эта характеристика тоже не идеальна.

Приведем пример NP-трудной оптимизационной задачи и соответствующего ей выпуклого многогранника, число прямоугольного покрытия для которого полиномиально относительно размерности.

Пример 5. В NP-трудной задаче о максимальной клике в произвольном (неполном) графе G на n вершинах требуется найти клику максимального размера (если таких клик несколько, то достаточно предъявить любую из них). Эта задача легко переформулируется в виде задачи линейной оптимизации на многограннике BQR_n . С этой целью установим взаимно-однозначное соответствие между вершинами графа и координатами c_{ii} , $1 \leq i \leq n$, вектора входа. Положим $c_{ii} = 1$ для всех $i \in [1, n]$, c_{ij} положим равными 0 для тех ребер, которые входят в граф G , и $c_{ij} = -2$ для отсутствующих ребер. Тогда, очевидно, максимум скалярного произведения $\langle c, x \rangle$, где $x \in \text{BQR}_n$, окажется равен размеру максимальной клики. Если же, кроме того, нам известны координаты x_{ii} , $i \in [1, n]$, вершины x многогранника BQR_n , в которой достигается максимум, то по их значениям легко определить вершины графа G , образующие максимальную клику.

Заметим, что если мы сместим одну из вершин многогранника BQR_n в произвольном направлении не более, чем на $\frac{1}{3n}$, то значение описанной выше целевой функции $\langle c, x \rangle$ в этой точке изменится не более, чем на $\frac{\sqrt{2n(n-1)+n}}{3n} < 1/2$. Так как на вершинах многогранника BQR_n наша целевая функция может принимать только целые значения, то такая пертурбация не сможет вывести в лидеры вершину, на которой функция не достигала максимального значения. Таким образом мы можем поступить со всеми вершинами. При этом максимум целевой функции изменится

менее, чем на $1/2$, что позволяет легко восстановить как размер максимальной клики, так и множество входящих в нее вершин. Более того, за счет малых смещений мы всегда можем добиться того, что все вершины окажутся в общем положении⁷, соответственно, сам новый многогранник станет симплицеальным. Остается воспользоваться тем, что число прямоугольного покрытия для симплицеального многогранника равно $O(d^2 \log_2 K)$, где d — размерность многогранника, а K — число его вершин [14]. То есть в нашем случае это число окажется равным $O(n^3(n+1)^2)$.

Список литературы

1. *Бондаренко В.А.* Полиэдральные графы и сложность в комбинаторной оптимизации. Ярославль: ЯрГУ, 1995. [*Bondarenko V.A.* Poliedralnye grafy i slozhnost v kombinatornoy optimizatsii. Yaroslavl: YarGU, 1995 (in Russian)].
2. *Бондаренко В.А., Максименко А.Н.* Геометрические конструкции и сложность в комбинаторной оптимизации. М.: ЛКИ, 2008. [*Bondarenko V.A., Maksimenko A.N.* Geometricheskie konstruksii i slozhnost v kombinatornoy optimizatsii. Moskva: LKI, 2008 (in Russian)].
3. *Бондаренко В.А., Николаев А.В.* Комбинаторно-геометрические свойства задачи о разрезе // Доклады Академии наук. Математика. 2013. Т. 452, № 2. С. 127–129. (English transl.: *Bondarenko V.A., Nikolaev A.V.* Combinatorial and Geometric Properties of the Max-Cut and Min-Cut Problems // *Doklady Mathematics*. 2013. V. 88, No 2. P. 516–517.)
4. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. (*Garey M.R. and Johnson D.S.* Computers and Intractability: A Guide to the Theory of NP-Completeness. A Series of Books in the Mathematical Sciences. San Francisco, Calif.: W. H. Freeman and Co., 1979.)
5. *Емеличев В.А., Ковалев М.М., Кравцов М.К.* Многогранники, графы, оптимизация. М.: Наука, 1981. (English transl.: *Yemelichev V.A., Kovalev M.M., Kravtsov M.K.* Polytopes, Graphs and Optimisation; translated by G.H. Lawden. Cambridge: Cambridge Univ. Press, 1984.)
6. *Колесов Е.В., Максименко А.Н.* Анализ одного алгоритма для задачи о назначениях // Заметки по информатике и математике. Ярославль: ЯрГУ, 2009. Вып. 1. С. 36–40. [*Kolesov E.V., Maksimenko A.N.* Analiz odnogo algoritma dlya zadachi o naznacheniyakh // *Zametki po informatike i matematike*. Yaroslavl: YarGU, 2009. Вып. 1. S. 36–40 (in Russian)].
7. *Максименко А.Н.* Комбинаторные свойства многогранника задачи о кратчайшем пути // Журнал вычислительной математики и математической физики. 2004. Т. 44. № 9. С. 1693–1696. (English transl.: *Maksimenko A.N.* Combinatorial properties of the polyhedron of the shortest path problem // *Comput. Math. Math. Phys.* 2004. V. 44, No. 9, P. 1611–1614.)
8. *Максименко А.Н.* k -смежностные грани булева квадратичного многогранника // Фундаментальная и прикладная математика. 2013. Т. 18, № 2. С. 95–103. [*Maksimenko A.N.*

⁷Точки множества X , $X \subset \mathbb{R}^d$, находятся в *общем положении*, если для любого подмножества $Y \subseteq X$, состоящего не более, чем из $d + 1$ точек, размерность его аффинной оболочки ровно на единицу меньше числа входящих в него точек.

- k-smezhnostnye grani buleva kvadraticnogo mnogogrannika // Fundamentalnaya i prikladnaya matematika. 2013. T. 18, № 2. S. 95–103 (in Russian)].
9. *Мощков М.Ю.* Об условных тестах // Доклады АН СССР. 1982. Т. 265, № 3. С. 550–552. (English transl.: *Moshkov M.Ju.* On conditional tests // Academy of Sciences Doklady. 1982. V. 265, No 3. P. 550–552.)
 10. *Bogomolov Yu., Fiorini S., Maksimenko A., Pashkovich K.* Small Extended Formulations for Cyclic Polytopes // arXiv:1401.8138
 11. *Conforti M., Cornuéjols G., and Zambelli G.* Extended formulations in combinatorial optimization // A Quarterly Journal of Operations Research. 2010. V. 8, No 1. P. 1–48.
 12. *Deza M.M., Laurent M.* Geometry of cuts and metrics. Springer, 1997.
 13. *Fiorini S., Massar S., Pokutta S., Tiwary H.R., de Wolf R.* Linear vs. semidefinite extended formulations: exponential separation and strong lower bounds // STOC. 2012. P. 95–106.
 14. *Fiorini S., Kaibel V., Pashkovich K., Theis D.O.* Combinatorial Bounds on Nonnegative Rank and Extended Formulations // Discrete Math. 2013. V. 313, No 1. P. 67–83.
 15. *Gaiha P., Gupta S.K.* Adjacent vertices on a permutohedron // SIAM Journal on Applied Mathematics. 1977. V. 32, No 2. P. 323–327.
 16. *Kaibel V.* Extended Formulations in Combinatorial Optimization // Optima. 2011. 85.
 17. *Kaibel V., Weltge S.* A Short Proof that the Extension Complexity of the Correlation Polytope Grows Exponentially // arXiv:1307.3543, 2013.
 18. *Maksimenko A.N.* A special place of Boolean quadratic polytopes among other combinatorial polytopes // arXiv:1408.0948
 19. *Rothvoss T.* The matching polytope has exponential extension complexity // STOC. 2014. P. 263–272.
 20. *Yannakakis M.* Expressing combinatorial optimization problems by linear programs // J. Comput. System Sci. 1991. V. 43, No 3. P. 441–466.

Characteristics of Complexity: Clique Number of a Polytope Graph and Rectangle Covering Number

Maksimenko A. N.

*P.G. Demidov Yaroslavl State University,
Sovetskaya str., 14, Yaroslavl, 150000, Russia*

Keywords: combinatorial optimization, convex polytopes, complexity of problems and algorithms, 1-skeleton of a polytope, clique number, extended formulations, facet-vertex incidence matrix, rectangle covering number

In the 1980s V.A. Bondarenko found that the clique number of the graph of a polytope in many cases corresponds to the actual complexity of the optimization problem on the vertices of the polytope. For an explanation of this phenomenon he proposed the theory of direct type algorithms. This theory asserts that the clique number of the graph of a polytope is the lower bound of the complexity of the corresponding problem in the so-called class of direct type algorithms. Moreover, it was argued that this class is wide enough and includes many classical combinatorial algorithms. In this paper we present a few examples, designed to identify the limits of applicability of this theory. In particular, we describe a modification of algorithms that is quite frequently used in practice. This modification takes the algorithms out of the specified class, while the complexity is not changed. Another, much closer to reality combinatorial characteristic of complexity is the rectangle covering number of the facet-vertex incidence matrix, introduced into consideration by M. Yannakakis in 1988. We give an example of a polytope with a polynomial (with respect to the dimension of the polytope) value of this characteristic, while the corresponding optimization problem is NP-hard.

Сведения об авторе:

Максименко Александр Николаевич,

Ярославский государственный университет им. П. Г. Демидова,
канд. физ.-мат. наук, науч. сотр. лаборатории «Дискретная и вычислительная
геометрия» им. Б. Н. Делоне